DTD Reference

Identity Manager 4.0.2

May 2014

Novell

Legal Notices

Novell, Inc. makes no representations or warranties with respect to the contents or use of this documentation, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc. makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

Any products or technical information provided under this Agreement may be subject to U.S. export controls and the trade laws of other countries. You agree to comply with all export control regulations and to obtain any required licenses or classification to export, re-export or import deliverables. You agree not to export or re-export to entities on the current U.S. export exclusion lists or to any embargoed or terrorist countries as specified in the U.S. export laws. You agree to not use deliverables for prohibited nuclear, missile, or chemical biological weaponry end uses. See the Novell International Trade Services Web page (http://www.novell.com/info/exports/) for more information on exporting Novell software. Novell assumes no responsibility for your failure to obtain any necessary export approvals.

Novell, Inc.
1800 South Novell Place
Provo, UT 84606
U.S.A.
www.novell.com

Online Documentation: To access the latest online documentation for this and other Novell products, see the Novell Documentation Web page (http://www.novell.com/documentation).

Novell Trademarks

For Novell trademarks, see the Novell Trademark and Service Mark list (http://www.novell.com/company/legal/trademarks/tmlist.html).

Third-Party Materials

All third-party trademarks are the property of their respective owners.

Contents

About This Guide

This guides is a reference to the document type definitions (DTD) that Identity Manager uses. The guide contains definitions for each of the elements used in Identity Manager. There are separate DTDs for different components of Identity Manager.

Audience

This guide is intended as a reference for Identity Manager consultants.

Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use the User Comments feature at the bottom of each page of the online documentation and enter your comments there.

Documentation Updates

For the most recent version of the *Identity Manager DTD Reference*, and the latest Identity Manager documentation, visit the Identity Manager Documentation Web site.

Documentation Conventions

In Novell documentation, a greater-than symbol (>) is used to separate actions within a step and items in a cross-reference path.

When a single pathname can be written with a backslash for some platforms or a forward slash for other platforms, the pathname is presented with a backslash. Users of platforms that require a forward slash, such as Linux or UNIX, should use forward slashes as required by your software.

1 DTD Overview

This guides is a reference for the Identity Manager document type definitions (DTD). There are separate DTDs for different components of Identity Manager:

What's New in Identity Manager 4.0.2

- Added do-add-resource, do-remove-resource, and do-start-workflow.
- Added Global Configuration Values section that provides information about GCVs that can be used by the driver to control functionality.
- Added DS-Object DTD section that provides information about the XML structure used to create objects in the Identity Manager drivers.

What's New in Identity Manager 4.0.1

Version 4.0.1 of DTD has no new features.

What's New in Identity Manager 3.6.1

Version 3.6.1 of DTD has no new features.

What's New in Identity Manager 3.6

- Added Jobs DTD section that provides introductory information about the XML structure used to create scheduled Jobs in Identity Manager.
- Added do-add-role and do-remove-role.
- Changed do-send-email-from-template to allow sending html content from policy.
- Made attribute `policy-dn` optional for token-generate-password.
- Changed do-find-matching-object to set a local variable if they encounter an error and provide additional information in the server log.
- Added `offset` and `offset-unit` attribute to token-convert-time.

- Added optional `default-value` attribute to token-map.
- Added optional `old-password` attribute to do-set-src-password and do-set-dest-password.

2 Filter DTD

An Identity Manager filter is primarily for controlling which object classes are synchronized and which attributes are synchronized for those object classes. Additionally, other behaviors of those classes and attributes within Identity Manager are controlled through the filter.

An Identity Manager filter consists of a top level <filter> element that contains a set of <filter-class> elements, each of which contains a set of <filter-attr> elements. The filter for a particular driver is stored in the DirXML-DriverFilter attribute on the DirXML-Driver object.

See "Filter Elements" on page 15 for a list of all the elements in the Filter DTD.

2.1 Filter Elements

Element	Description
filter	Filter for an Identity Manager driver.
filter-attr	Behavior of an attribute for a particular object class.
filter-class	Behavior of an object class.

filter

Consists of a set of <filter-class> elements that describe the object classes used by a particular instance of an Identity Manager driver.

Example

```
<filter>
   <filter-class  class-name="User"
      subscriber="sync"
      publisher="sync">

      <filter-attr attr-name="CN"
       subscriber="sync"
       publisher="ignore"
       merge-authority="none"/>

      <filter-attr attr-name="Surname"
       subscriber="sync"/>

  <filter-attr attr-name="Given name"
       subscriber="sync"/>

  <filter-attr attr-name="Internet EMail Address"
       publisher="sync"
       publisher-optimize-modify="false"/>

  <filter-attr attr-name="Login Disabled"
       subscriber="notify"/>
   </filter-class>
   <filter-class class-name="Group"
         subscriber="sync"
         publisher="sync">
         publisher-create-homedir="false">

      <filter-attr attr-name="CN"
       subscriber="sync"
       merge-authority="none"/>

  <filter-attr attr-name="Member"
       subscriber="sync"
       publisher="sync"
       merge-authority="publisher"
       publisher-optimize-modify="false"/>
   </filter-class>
</filter>
```

Allowed Content

Element	Description
filter-class	Behavior of an object class.

Attributes

None

Content Rule

(filter-class*)

Parent Elements

None

filter-attr

Describes an attribute of the enclosing <filter-class> that is used by a particular instance of an Identity Manager driver. The attr-name attribute specifies the name of an attribute in eDirectory.The publisher and subscriber attributes control whether this attribute is synchronized on the respective channels according to the following table. If the channel setting for the object class as a whole is ignored, then the setting for individual attributes is ignored.

Remarks

The publisher and subscriber attributes control whether this attribute is synchronized on the respective channels according to the following table. If the channel setting for the object class as a whole is ignored, then the setting for individual attributes is ignored.

Value	Description
ignore	Changes to this attribute are not reported or automatically synchronized.
notify	Changes to this attribute are reported but not automatically synchronized.
sync	Changes to this attribute are reported and automatically synchronized.
reset	Changes to this attribute are reported and triggers the attribute to be automatically reset to the values from the other channel. It is illegal for both Publisher and Subscriber to reset for the same attribute.

The merge-authority attribute controls the behavior of the attribute during a merge operation according to the following table:

Value	Behavior	Valid
default	1. If an attribute is not being synchronized in either channel, then no merging occurs. 2. If an attribute is being synchronized in one channel and not the other, then all existing values on the destination for that channel are removed and replaced with the values from the source for that channel. If the source has multiple values and the destination can only accommodate a single value, then only one of the values is used on the destination side, although it is undefined which of those values are used. 3. If an attribute is being synchronized in both channels and both sides can accommodate multiple values, then each side ends up with the union of values present on either side. 4. If an attribute is being synchronized in both channels and both sides can accommodate only a single value, the application ends up with the value from eDirectory unless there is no value in eDirectory. In this case eDirectory ends up with the value from the application (if any). 5. If an attribute is synchronized in both channels and only one side can accommodate multiple values then the single-valued side's value is added to the value from the multiple-value side if it is already there. If there is no value on the single-valued side one of the values (undefined) is added to the single-valued side.	Always
edir	Has the same behavior as the default if the attributes are synchronized on the Subscriber channel and not on the Publisher channel.	When synchronizing or notifying on the Subscriber channel
app	Has the same behavior as the default if the attributes are synchronized on the Publisher channel and not on the Subscriber channel.	When synchronizing or notifying on the Publisher channel
none	No merging occurs regardless of synchronization.	Always

The publisher-optimize-modify attribute controls whether or not changes to this attribute are examined on the Publisher channel to determine the minimal change needed in the Identity Vault.

Example

See <filter>.

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
attr-name	CDATA Name of the attribute.	#REQUIRED
merge-authority	default | edir | app | none Flag that controls how this attribute is merged.	default
publisher	ignore | notify | sync | reset Flag that controls Publisher channel synchronization.	ignore
publisher-optimize-modify	true | false Flag that controls optimization of modifications on the Subscriber channel.	true
subscriber	ignore | notify | sync | reset Flag that controls Subscriber channel synchronization.	ignore

Content Declaration

Empty

Parent Elements

Element	Description
filter-class	Behavior of an object class.

filter-class

Describes an object class that is used by a particular instance of an Identity Manager driver. The class-name attribute specifies the name of an effective (that is, structural or base) class in eDirectory and only applies to objects that have that particular base class.

Remarks

The Publisher and Subscriber attributes control whether this class is synchronized on the respective channels.

Value	Description
ignore	Changes to the objects of this class are not reported or automatically synchronized.
sync	Changes to the objects of this class are reported and automatically synchronized.

The publisher-track-template-member attribute controls whether or not the Publisher channel maintains the Member of Template attribute when it creates objects from a template. The publisher-create-homedir attribute controls whether or not a NetWare home directory is automatically created when a User is created with the Home Directory attribute populated.

Example

See <filter>.

Allowed Content

Element	Description
filter-attr	Behavior of an attribute for a particular object class.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA	#REQUIRED
	Name of the object class.	
publisher	ignore | sync	ignore
	Flag that controls Publisher channel synchronization.	
publisher-create-homedir	true | false	true
	Flag that controls automatic creation of home directories.	
publisher-track-template-member	true | false	false
	Flag that controls the tracking of objects.	
subscriber	ignore | sync	ignore
	Flag that controls Publisher channel synchronization.	

Content Rule

filter-attr

Parent Elements

Element	Description
filter	Filter for an Identity Manager driver.

3 NDS DTD

The NDS document type definition file (`nds.dtd`) defines the schema of the XML documents that the Identity Manager engine can process. XML documents that do not conform to this schema generate errors.

The `nds.dtd` file defines the following:

- Input and output commands and events (such as add, delete, modify, and rename) that can be performed on entries and the data that must be included with each.
- Driver initialization operations (such as authentication information, driver filter, configuration options, and state) for the driver shim, publisher shim, and subscriber shim and the data that these operations require.
- Schema operations for defining class and attribute definitions.
- Rules for schema mapping, matching, creation, and placement.

Remember the following when reading a DTD file:

Marker	Meaning
?	0 or 1 of these can be included.
+	1 or more of these must be included.
*	0 or more of these can be included.
CDATA	Character data.
PCDATA	Parsed character data.
<!	Beginning of an element, entity, or attribute definition.
>	End of an element, entity, or attribute definition.

See "NDS DTD Elements" on page 23 for a list of all of the elements in the NDS DTD.

3.1 NDS DTD Elements

Element	Description
add	Adds an object when an add event occurs.
add-association	Adds an association.
add-attr	Adds an attribute.
add-value	Adds values.

Element	Description
allow-attr	Allows an attribute in the filter.
allow-class	Allows a class in the filter.
app-name	Names in the application namespace.
association	Unique key of the application object.
attr	Current state of an attribute.
attr-def	Schema attribute definition.
attr-name	Maps an attribute name.
attr-name-map	Top-level element for Schema Mapping policies.
authentication-info	Information for connecting and authenticating to the application.
check-object-password	Checks the password against an eDirectory object.
check-password	Checks the password against an eDirectory driver object.
class-def	Schema class definition.
class-name	Maps a class name.
component	Component of a structured attribute.
config-object	eDirectory object to use for additional configuration data.
contact	Point of contact for the originating product.
delete	Deletes an object when a delete event occurs.
driver-config	Driver-specific Driver Shim configuration options.
driver-filter	Publication and Subscription class and attribute event filter.
driver-options	Driver-specific Driver Shim configuration options.
driver-state	Driver-specific state information.
get-named-password	Retrieves a named password for a driver.
init-params	Initialization parameters for the DriverShim, SubscriptionShim, or PublicationShim.
input	Input events or commands.
instance	Current state of an instance of an object.
modify	Modifies an object when a modify event occurs.
modify-association	Modifies an association command.
modify-attr	Modifies an attribute.
modify-password	Modifies an object password when a modify event for a password occurs.
move	Moves an object when a move event occurs.

Element	Description
nds	Top-level element for all Identity Manager and Driver communication.
nds-name	Name in the eDirectory namespace.
new-name	The new name of a renamed object.
old-password	The old authentication password.
operation-data	The operation adds additional custom data.
output	Results of events or commands.
parent	The parent container of an object.
password	The authentication password.
product	The product from which the document originated.
publisher-options	Driver-specific PublicationShim configuration options.
publisher-state	Driver PublicationShim state information.
query	Query command.
query-ex	Query command with result count limit.
query-schema	Query schema command.
query-token	Opaque handle for query-ex commands.
rread-attr	Returns specified object attribute values.
read-parent	Returns the object parent container.
remove-all-values	Removes all attribute values.
remove-association	Removes an association.
remove-value	Removes specified attribute values.
rename	Renames an object when a rename event occurs.
schema-def	Schema definition.
search-attr	Query search attribute value filter.
search-class	Query search class filter.
server	The authentication server.
source	The source or creator of the document.
status	Status of the processing of a command or event.
subscriber-options	Driver-specific SubscriptionShim configuration options.
subscriber-state	Driver SubscriptionShim state information.
sync	Resynchronization or migrate event.
user	The authentication user name
value	The attribute value.

add

Used as an event notification from the PublicationShim to Identity Manager when an object is added in the application. When it is used as a notification, an <association> is required. It is also used as a command from Identity Manager to the SubsciptionShim to add an object in the application.

Remarks

<add> contains an <add-attr> for each attribute of the object added.

<add> might contain a <password> for the object added.

A response to <add> should be a <status> indicating whether or not the <add> was processed successfully. When used as a command, <add> should also return an <add-association> that contains the unique key for the newly added object. The dest-dn and dest-entry-id attributes of the <add-association> should be set to the src-dn and src-entry-id of the <add>.

Example

```
<add class-name="User" src-dn="\Sam">
        <association>1012</association>
        <add-attr attr-name="cn">
                <value>Sam</value>
        </add-attr>
        <add-attr attr-name="Surname">
                <value>Jones</value>
        </add-attr>
        <add-attr attr-name="Given Name">
                <value>Sam</value>
        </add-attr>
        <add-attr attr-name="Telephone Number">
                <value>555-1212</value>
        </add-attr>
</add>
```

Allowed Content

Element	Description
association	Unique key of the application object.
add-attr	Add attribute.
password	The authentication password.
operation-data	The operation adds additional custom data.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA The name of the base class of the object. The class name is mapped between the application and eDirectory namespaces by the Schema Mapping policy so that Identity Manager sees the name in the eDirectory namespace and a driver sees the name in the application namespace.	#REQUIRED
dest-dn	CDATA The distinguished name of the target object in the namespace of the receiver. Should be left empty for event notifications. Filled in by the Placement policy on commands.	#IMPLIED
dest-entry-id	CDATA The entry ID of the target object in the namespace of the receiver. Reserved. Should be ignored by the driver.	#IMPLIED
event-id	CDATA An identifier used to tag the results of an event or command. Should be copied to the event-id attribute of the resulting <status> and <add-association> elements.	#IMPLIED
qualified-src-dn	CDATA The qualified version of src-dn. Only used for describing objects from eDirectory.	#IMPLIED
src-dn	CDATA The distinguished name of source object that generated the event in the namespace of the sender. Should be copied to the dest-dn attribute of the resulting <add-association> for commands.	#IMPLIED

Attribute	Possible Values	Default Value
src-entry-id	CDATA The entry ID of source object that generated the event in the namespace of the sender. Should be copied to the dest-entry-id attribute of the resulting <add-association> for commands.	#IMPLIED
template-dn	CDATA The distinguished name of a template in the receiver's namespace to use as a basis for creating the object. Filled in by the Create policy for commands. Drivers only need to implement this if it makes sense for the application.	#IMPLIED
timestamp	CDATA Reserved. Should be ignored by the driver.	#IMPLIED

Content Rule

(association ? , add-attr * , password ? , operation-data ?)

Parent Elements

Element	Description
input	Input events or commands.

add-association

Used to return the unique key of an object added as the result of an <add> command.

Example

```
<add-association dest-dn="\Users\Samuel" dest-entry-id="33974">
   {BC3E7155-CDF9-d311-9846-0008C76B16C2}
</add-association>
```

Allowed Content

#PCDATA

Element	Description
operation-data	Operation adds additional custom data.

Attributes

Attribute	Possible Values	Default Value
dest-dn	CDATA The distinguished name of the target object in the namespace of the receiver. Should be set to the src-dn of the <add>.	#REQUIRED
dest-entry-id	CDATA The entry id of the target object in the namespace of the receiver. Should be set to the src-entry-id of the <add>.	#IMPLIED
event-id	CDATA An identifier used to tag the results of an event or command. Should be set to the event-id of the <add>.	#IMPLIED

Content Rule

(#PCDATA | operation-data) *

Parent Elements

Element	Description
input	Input events or commands.
output	Results of events or commands.

add-attr

Used to specify the attribute values for an <add> operation or event. Each <add-attr> should contain at least one <value>

Example

See <add>.

Allowed Content

Element	Description
value	The attribute value.

Attributes

Attribute	Possible Values	Default Value
attr-name	CDATA The name of the attribute. The name is mapped between the application and eDirectory namespaces by the Schema Mapping policy so that Identity Manager sees the name in the eDirectory namespace and a driver sees the name in the application namespace. The mapping rule uses the class name attribute of the enclosing command or event to determine which class to use for mapping the attribute name.	#REQUIRED
enforce-password-policy	true | false Reserved. Should be ignored by the driver.	#IMPLIED

Content Rule

(value +)

Parent Elements

Element	Description
add	Adds an object when an add event occurs.

add-value

Used to specify values added to the attribute specified in the enclosing <modify-attr>. A driver should gracefully ignore an <add-value> for a value that already exists and continue to process the remainder of the enclosing <modify>.

Example

See <modify>.

Allowed Content

Element	Description
value	The attribute value.

Attributes

None

Parent Elements

Element	Description
modify-attr	Modifies an attribute.

allow-attr

Used to specify attributes that are allowed in the event filter for the class specified in the enclosing <allow-class>.

Example

See <init-params>.

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
attr-name	CDATA The name of the attribute. The name is mapped between the application and eDirectorynamespaces by the Schema Mapping policy so that Identity Manager sees the name in the eDirectory namespace and a driver sees the name in the application namespace. The mapping rule uses the class name attribute of the enclosing command or event to determine which class to use for mapping the attribute name.	#REQUIRED
is-sensitive	true | false If true, specifies that the attribute values referenced by the <allow-attr> element contain sensitive data that should be suppressed in trace information.	false

Content Declaration

Empty

Parent Elements

Element	Description
allow-class	Allow a class in the filter.

allow-class

Used to specify classes that are allowed in the event filter specified by the enclosing <driver-filter>.

Example

See <init-params>.

Allowed Content

Element	Description
allow-attr	Allow an attribute in the filter.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA The name of the base class of the object. The class name is mapped between the application and eDirectory namespaces by the Schema Mapping policy so that Identity Manager sees the name in the eDirectory namespace and a driver sees the name in the application namespace.	#REQUIRED

Content Rule

(allow-attr) *

Parent Elements

Element	Description
driver-filter	Publication and Subscription class and attribute event filter.

app-name

Used to specify a class or attribute name in the application namespace.

Example

See <attr-name-map>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
attr-name	Maps an attribute name.
class-name	Maps a class name.

association

Used to specify the unique key of an application object that is the source of an event notification from the PublicationShim to Identity Manager, the target of a command sent form Identity Manager to the SubscriptionShim, or the base object of a <query> sent to the SubscriptionShim.

Example

```
<association state="associated">
   {B43E7155-CDF9-d311-9846-0008C76B16C2}
</association>
```

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
state	not-associated | associated | disabled | migrate | pending | manual Reserved: Should be ignored by the driver	#IMPLIED

Content Rule

#PCDATA

Parent Elements

Element	Description
add	Adds an object when an add event occurs.
check-object-password	Checks the password against an eDirectory object.
delete	Deletes an object when a delete event occurs.
instance	Current state of an instance of an object.
modify	Modifies an object when a modify event occurs.
modify-association	Modifies an association command.
modify-password	Modifies an object password when a modify event for a password occurs.
move	Moves an object when a move event occurs.
parent	The parent container of an object.
query	Query command.

Element	Description
query-ex	Query command with a result count limit.
rename	Renames an object when a rename event occurs.
sync	Resynchronization or migrate event.

attr

Used to specify the attribute values for the object specified by the enclosing <instance>. Each <attr> should contain at least one <value>.

Example

See <instance>.

Allowed Content

Element	Description
value	The attribute value.

Attributes

Attribute	Possible Values	Default Value
attr-name	CDATA The name of the attribute. The name is mapped between the application and eDirectory namespaces by the Schema Mapping policy so that Identity Manager sees the name in the eDirectory namespace and a driver sees the name in the application namespace. The mapping rule uses the class name attribute of the enclosing command or event to determine which class to use for mapping the attribute name.	#REQUIRED

Content Rule

(value *)

Parent Elements

Element	Description
instance	Current state of an instance of an object.

attr-def

Used to specify a schema attribute for the class specified by the enclosing <class-def>.

Example

See <schema-def>.

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
asn1id	CDATA The ASN.1 Object ID of the attribute.	#IMPLIED
attr-name	CDATA The name of the attribute.	#REQUIRED
case-sensitive	true | false Whether or not the attribute is case sensitive.	false
multi-valued	true | false Whether or not the attribute can hold more than one value	true
naming	true | false Whether or not the attribute can be used as part of the RDN of an object of the enclosing class.	false
read-only	true | false Whether or not the attribute is read-only.	false
required	true | false Whether or not the attribute is required by an object of the enclosing class.	false
type	string | teleNumber | int | state | counter | dn | interval | octet | time | structured The data type of the attribute.	string

Content Declaration

Empty

Parent Elements

Element	Description
class-def	Schema class definition.

attr-name

Used to specify a mapping between an attribute name in the eDirectory namespace and the application namespace.

Example

See <attr-name-map>.

Allowed Content

Element	Description
nds-name	Name in the eDirectory namespace. The names specified must be unique for the given class.
app-name	Name in the application namespace. The names specified must be unique for the given class.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA The name of the class that this attribute name mapping is for in the eDirectory namespace. If missing or blank then mapping is considered generic and applies to all classes that don't have a class-specific mapping.	#IMPLIED

Content Rule

(nds-name , app-name)

Parent Elements

Element	Description
attr-name-map	Top-level element for Schema Mapping policies.

attr-name-map

The top-level (document) element for Schema Mapping policies. Schema Mapping policies are stored in the DirXML-XmlData attribute of a DirXML-Rule object that is pointed to by the DirXML-MappingRule attribute of a DirXML-Driver object.

Remarks

<attr-name-map> contains <attr-name> and <class-name> elements that specify a one-to-one mapping between class and attribute names in eDirectory and the application namespace. Schema Mapping policies are applied to map from the eDirectory namespace to the application namespace whenever XML is sent or returned from Identity Manager to the driver and before the Output transform is applied.

Schema Mapping policies are applied to map from the application namespace to the eDirectory namespace whenever XML is sent or returned from driver to Identity Manager after the Input Transform policy is applied.

Schema Mapping policies try to map the <class-name> and <attr-name> attributes of all elements in the document. The <class-name> to map an attribute name is found by looking the nearest ancestor element with a <class-name> attribute.

Example

```
<attr-name-map>
    <!-- map eDirectory class User application class inetOrgPerson
-->
        <class-name>
                <nds-name>User</nds-name>
                <app-name>inetOrgPerson</app-name>
        </class-name>
        <!-- map NDS attribute Given Name to application attribute givenName for
class User -->
        <attr-name class-name="User">
                <nds-name>Given Name</nds-name>
                <app-name>givenName</app-name>
        </attr-name>
        <!-- map NDS attribute Surname to application attribute sn for all classes
-->
        <!-- that don't have a class-specific mapping -->
        <attr-name>
                <nds-name>Surname</nds-name>
                <app-name>sn</app-name>
        </attr-name>
</attr-name-map>
```

Allowed Content

Element	Description
attr-name	Maps an attribute name.
class-name	Maps a class name.

Attributes

None

Content Rule

(attr-name | class-name) *

Parent Elements

None

authentication-info

Used to specify the parameters needed for the driver to connect to and authenticate to an application server.

Example

See <init-params>.

Allowed Content

Element	Description
server	The authentication server.
user	The authentication username.
password	The authentication password.

Attributes

None

Content Rule

(server ? , user ? , password ?)

Parent Elements

Element	Description
init-params	Initialization parameters for the DriverShim, SubscriptionShim, or PublicationShim.

check-object-password

Used to validate a password against an eDirectory object. A <status> result is returned indicating success or failure.

Remarks

An <association> element, a dest-dn attribute, or a dest-entry-id attribute is used to specify the eDirectory object against which the password is to be checked.

If the eDirectory object's Login Disabled attribute is set to true, then the <status> indicates an error even if the password is correct.

Example

```
<check-object-password dest-dn="container\object">
   abdc1234
</check-object-password>
```

Allowed Content

Element	Description
association	Unique key of the application object.
password	The authentication password.
operation-data	Operation additional custom data.

Attributes

Attribute	Possible Values	Default Value
dest-dn	CDATA	#IMPLIED
	The distinguished name of the target object in the namespace of the receiver.	
dest-entry-id	CDATA	#IMPLIED
	The entry ID of the target object in the namespace of the receiver.	
event-id CDATA	CDATA	#IMPLIED
An identifier used to tag the results of an event or command.	An identifier used to tag the results of an event or command.	

Content Rule

(association ? , password , operation-data ?)

Parent Elements

Element	Description
input	Input events or commands.

check-password

Used to validate a password against the eDirectory driver object. A <status> result is returned indicating success or failure.

Example

```
<check-password>abdc1234</check-password>
```

Allowed Content

#PCDATA

Element	Description
operation-data	The operation adds additional custom data.

Attributes

None

Content Rule

(#PCDATA | operation-data) *

Parent Elements

Element	Description
input	Input events or commands.

class-def

Used to specify a schema class the enclosing <schema-def>.

Example

See <schema-def>.

Allowed Content

Element	Description
attr-def	Schema attribute definition.

Attributes

Attribute	Possible Values	Default Value
asn1id	CDATA The ASN.1 Object ID of the class.	#IMPLIED
class-name	CDATA The name of the schema class.	#REQUIRED
container	true | false Whether or not an object of this class can be a container for other objects.	false

Content Rule

(attr-def) *

Parent Elements

Element	Description
schema-def	Schema definition.

class-name

Used to specify a mapping between a class name in the eDirectory namespace and the application namespace.

Example

See <attr-name-map>.

Allowed Content

Element	Description
nds-name	Name in the eDirectory namespace. The names specified must be unique to this <class-name>.
app-name	Name in the eDirectory namespace. The names specified must be unique to this <class-name>.

Attributes

None

Content Rule

(nds-name , app-name)

Parent Elements

Element	Description
attr-name-map	Top-level element for Schema Mapping policies.

component

Used to specify an individual field of the enclosing <value> if the data type of the value is structured.

Example

See <value>.

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
association-ref	CDATA The association value (application object unique key) of the object being referenced by this component. This is required on all components that refer to other objects when the component is part of a notification event from the driver. This exists on all components that refer to other objects when the component is part of a command from Identity Manager if the referenced object has an established association in eDirectory.	#IMPLIED
name	CDATA The name of the component. This is specific to individual attribute syntaxes. See <value>.	#REQUIRED

Content Rule

(#PCDATA)

Parent Elements

Element	Description
value	The attribute value.

config-object

Used to specify objects and attributes where additional configuration information is obtained.

Remarks

During driver startup, the contained <query> is processed and the resulting <instance> element replaces the <config-object> in the <init-params> passed to the DriverShim.init(), SubscriptionShim.init(), and PublicationShim.init() methods.

Example

See <init-params>.

Allowed Content

Element	Description
query	Query command.

Attributes

Attribute	Possible Values	Default Value
display-name	CDATA The name to display in the interface generated by ConsoleOne.	#IMPLIED

Content Rule

(query)

Parent Elements

None

contact

Used to specify the point of contact for the creator of the enclosing document.

Example

See <nds>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

#PCDATA

Parent Elements

Element	Description
source	The source or creator of the document.

delete

Used as an event notification from the PublicationShim to Identity Manager when an object is deleted in the application. When used as a notification, an <association> is required. Also used as a command from Identity Manager to the SubsciptionShim to delete an object in the application. When used as a command, an <association> is required and is the unique key of the object to delete.

Remarks

A response to <delete> should be a <status> indicating whether or not the <delete> was processed successfully.

Example

```
<delete class-name="User" src-dn="\Sam">
        <association>1012</association>
</delete>
```

Allowed Content

Element	Description
association	Unique key of the application object.
operation-data	The operation adds additional custom data.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA The name of the base class of the object. The class name is mapped between the application and eDirectory namespaces by the Schema Mapping policy so that Identity Manager sees the name in the eDirectory namespace and a driver sees the name in the application namespace.	#IMPLIED
dest-dn	CDATA The distinguished name of the target object in the namespace of the receiver. Should be left empty for event notifications.	#IMPLIED

Attribute	Possible Values	Default Value
dest-entry-id	CDATA The entry ID of the target object in the namespace of the receiver. Should be left empty for event notifications.	#IMPLIED
event-id	CDATA An identifier used to tag the results of an event or command.	#IMPLIED
qualified-src-dn	CDATA The qualified version of src-dn. Only used for describing objects from eDirectory.	#IMPLIED
src-dn	CDATA The distinguished name of source object that generated the event in the namespace of the sender.	#IMPLIED
src-entry-id	CDATA The entry ID of the source object that generated the event in the namespace of the sender. Reserved. Should be ignored by the driver.	#IMPLIED
timestamp	CDATA Reserved. Should be ignored by the driver.	#IMPLIED

Content Rule

(association ? , operation-data ?)

Parent Elements

Element	Description
input	Input events or commands.

driver-config

Used to specify driver-specific configuration options. It is the top-level element in the XML stored in the DirXML-ShimConfigInfo attribute of the DirXML-Driver object in eDirectory.

Remarks

The enclosed <driver-options>, <subscriber-options>, and <publisher-options> can each contain any number of <config-object> and driver-defined elements. The driver-defined elements might each contain text data.

Each driver-defined element can have a type attribute. The type attribute can specify that the element refers to a named password by assigning the value password-ref to the attribute. A named password reference is replaced at runtime with the actual value of a named password set using the Identity Manager administration facilities.

In the Identity Manager administration interface, each driver defined element is displayed as an edit control that can edit the content of the element. Each <config-object> is displayed as a single valued dn control that allows the selection of a dn to fill in the dest-dn of enclosed <query>. Each control is labeled with the value of the display-name attribute if it exists or with the tag name if it does not exist.

Example

```
<driver-config name="Netscape DirXML Driver">
        <driver-options>
                <display-method display-name="Debug Output (0-none,
1-Window, 2-DSTrace)">1</display-method>
        </driver-options>
        <subscriber-options>
            <config-object display-name="Super driver configuration
data">
               <query dest-dn="novell/Driver Set/Super
Driver/Config Object" scope="entry" event-id="config1">
                  <read-attr attr-name="Some Attribute"/>
                  <read-attr attr-name="XmlData" type="xml"/>
                </query>
             </config-object>
        </subscriber-options>
        <publisher-options>
           <pollRate display-name="Poll rate in seconds">5</pollRate>
                <changeLogSuffix display-name="Netscape changelog
suffix">cn=changelog</changeLogSuffix>
               <changeLogBegin display-name="Starting changelog (1-First,2-New, 3-
Continue)">2</changeLogBegin>
        </publisher-options>
</driver-config>
```

Allowed Content

Element	Description
driver-options	Driver-specific DriverShim configuration options.
subscriber-options	Driver-specific SubscriptionShim configuration options.
publisher-options	Driver-specific PublicationShim configuration options.

Attributes

Attribute	Possible Values	Default Value
name	CDATA Human readable name of the driver shim.	#IMPLIED

Content Rule

(driver-options ? , subscriber-options ? , publisher-options ?)

Parent Elements

None

driver-filter

Used to specify the event filter that is being used by a particular channel. It is generated from the DirXML-DriverFilter attribute on the DirXML-Subscriber or DirXML-Publisher object.

Example

See <init-params>.

Allowed Content

Element	Description
allow-class	Allows a class in the filter.

Attributes

Attribute	Possible Values	Default Value
type	publisher | subscriber Specifies the channel that the filter is for.	#IMPLIED

Content Rule

(allow-class) *

Parent Elements

Element	Description
init-params	Initialization parameters for a DriverShim, SubscriptionShim, or PublicationShim.

driver-options

Used to specify driver-specific configuration options. It comes from the DirXML-ShimConfigInfo attribute of the DirXML-Driver object in eDirectory.

Example

See <driver-config>.

Allowed Content

ANY

Attributes

None

Content Rule

ANY

Parent Elements

Element	Description
driver-config	Driver specific DriverShim configuration options.
init-params	Initialization parameters for a DriverShim, SubscriptionShim, or PublicationShim.

driver-state

Used specify driver specific state information.

Example

See <init-params>.

Allowed Content

ANY

Attributes

None

Content Rule

ANY

Parent Elements

Element	Description
init-params	Initialization parameters for a DriverShim, SubscriptionShim, or PublicationShim.

get-named-password

Used to retrieve a named password for a driver. A <status> result is returned indicating success or failure. If the status is success, then a <password> element is also returned containing the password value. The content of <get-namedpassword> is the name or key of the password that is retrieved.

Example

```
<get-named-password event-id="gnp37">
   web-password
</get-named-password>
```

Allowed Content

#PCDATA

Element	Description
operation-data	The operation adds additional custom data.

Attributes

Attribute	Possible Values	Default Value
event-id	CDATA An identifier used to tag the results of an event or command.	#IMPLIED

Content Rule

(#PCDATA | operation-data) *

Parent Elements

Element	Description
input	Input events or commands.

init-params

Used to specify initialization parameters for a DriverShim, SubscriptionShim, or PublicationShim.

Remarks

<init-params> is also included in any <output> or <input> from the driver to Identity Manager, which instructs Identity Manager to store the contents of the enclosed <driver-state>, <subscriber-state>, and <publisher-state> into the DirXML-DriverStorage attribute of the DirXML-Driver object in eDirectory. The states are included in the <init-params> sent to the corresponding init() function when a driver, subscriber, or publisher is started.

Example

```
<!-- for DriverShim.init() -->
<init-params src-dn="\MY_TREE\MyOrg\MyDriverSet\MyDriver">
        <authentication-info>
                <server>localhost</server>
                <user>Fred</user>
                <password>foobar</password>
        </authentication-info>
        <driver-options>
                <!-- some driver defined driver options -->
        </driver-options>
        <driver-state>
                <!-- some driver defined driver state -->
        </driver-state>
</init-params>
<!-- for SubscriptionShim.init() -->
<init-params src-dn="\MY_TREE\MyOrg\MyDriverSet\MyDriver\Subscriber">
        <authentication-info>
                <server>localhost</server>
                <user>Fred</user>
                <password>foobar</password>
        </authentication-info>
        <driver-filter type="subscriber">
                <allow-class class-name="User">
                        <allow-attr attr-name="Telephone Number"/>
                        <allow-attr attr-name="CN"/>
                        <allow-attr attr-name="Surname"/>
                        <allow-attr attr-name="Given Name"/>
                        <allow-attr attr-name="Description"/>
                        <allow-attr attr-name="Title"/>
                        <allow-attr attr-name="Postal Address"/>
                        <allow-attr attr-name="GUID"/>
                        <allow-attr attr-name="Full Name"/>
                </allow-class>
                <allow-class class-name="Organizational Unit">
                        <allow-attr attr-name="OU"/>
                </allow-class>
                <allow-class class-name="Organizational">
                        <allow-attr attr-name="O"/>
                </allow-class>
        </driver-filter>
        <subscriber-options>
                <!-- some driver defined subscriber options -->
        </subscriber-options>
        <subscriber-state>
                <!-- some driver defined subscriber state -->
        </subscriber-state>
</init-params>
<!-- for PublicationShim.init() -->
<init-params src-dn="\MY_TREE\MyOrg\MyDriverSet\MyDriver\Publisher">
        <authentication-info>
```

```
                <server>localhost</server>
                <user>Fred</user>
                <password>foobar</password>
        </authentication-info>
        <driver-filter type="publisher">
                <allow-class class-name="User">
                        <allow-attr attr-name="Telephone Number"/>
                        <allow-attr attr-name="CN"/
                        <allow-attr attr-name="Surname"/>
                        <allow-attr attr-name="Given Name"/>
                        <allow-attr attr-name="Description"/>
                        <allow-attr attr-name="Title"/>
                        <allow-attr attr-name="Postal Address"/>
                        <allow-attr attr-name="GUID"/>
                        <allow-attr attr-name="Full Name"/>
                </allow-class>
                <allow-class class-name="Organizational Unit">
                        <allow-attr attr-name="OU"/>
                </allow-class>
                <allow-class class-name="Organizational">
                        <allow-attr attr-name="O"/>
                </allow-class>
        </driver-filter>
        <publisher-options>
                <!-- some driver defined publisher options -->
        </publisher-options>
        <publisher-state>
                <!-- some driver defined publisher state -->
        </publisher-state>
</init-params>
<!-- for DriverShim.getSchema() -->
<init-params>
        <authentication-info>
                <server>localhost</server>
                <user>Fred</user>
                <password>foobar</password>
        </authentication-info>
        <driver-filter type="subscriber">
                <allow-class class-name="User">
                        <allow-attr attr-name="Telephone Number"/>
                        <allow-attr attr-name="CN"/>
                        <allow-attr attr-name="Surname"/>
                        <allow-attr attr-name="Given Name"/>
                        <allow-attr attr-name="Description"/>
                        <allow-attr attr-name="Title"/>
                        <allow-attr attr-name="Postal Address"/>
                        <allow-attr attr-name="GUID"/>
                        <allow-attr attr-name="Full Name"/>
                </allow-class>
                <allow-class class-name="Organizational Unit">
                        <allow-attr attr-name="OU"/>
                </allow-class>
                <allow-class class-name="Organizational">
                        <allow-attr attr-name="O"/>
                </allow-class>
        </driver-filter>
        <driver-filter type="publisher">
                <allow-class class-name="User">
                        <allow-attr attr-name="Telephone Number"/>
                        <allow-attr attr-name="CN"/>
                        <allow-attr attr-name="Surname"/>
                        <allow-attr attr-name="Given Name"/>
                        <allow-attr attr-name="Description"/>
                        <allow-attr attr-name="Title"/>
                        <allow-attr attr-name="Postal Address"/>
                        <allow-attr attr-name="GUID"/>
                        <allow-attr attr-name="Full Name"/>
                </allow-class>
                <allow-class class-name="Organizational Unit">
                        <allow-attr attr-name="OU"/>
```

```
                </allow-class>
                <allow-class class-name="Organizational">
                        <allow-attr attr-name="O"/>
                </allow-class>
        </driver-filter>
        <driver-options>
                <!-- some driver defined driver options -->
        </driver-options>
        <subscriber-options
                <!-- some driver defined subscriber options -->
        </subscriber-options>
        <publisher-options>
                <!-- some driver defined publisher options -->
        </publisher-options>
        <driver-state>
                <!-- some driver defined driver state -->
        </driver-state>
        <subscriber-state>
                <!-- some driver defined subscriber state -->
        </subscriber-state>
        <publisher-state>
                <!-- some driver defined publisher state -->
        </publisher-state>
</init-params>
```

Allowed Content

Element	Description
authentication-info	Information for connecting and authenticating to the application.
driver-filter	Publication and Subscription class and attribute event filter.
driver-options	Driver-specific DriverShim configuration options.
subscriber-options	Driver-specific SubscriptionShim configuration options.
publisher-options	Driver-specific PublicationShim configuration options.
driver-state	Driver-specific state information.
subscriber-state	Driver SubscriptionShim state information.
publisher-state	Driver PublicationShim state information.
operation-data	The operation adds additional custom data.

Attributes

Attribute	Possible Values	Default Value
src-dn	CDATA The distinguished name of DirXML-Driver, DirXML-Publisher, or DirXML-Subscriber.	#IMPLIED

Content Rule

(authentication-info ? , driver-filter ? , driver-options ? , subscriber-options ? , publisher-options ? , driver-state ? , subscriber-state ? , publisher-state ? , operation-data ?)

Parent Elements

Element	Description
input	Input events or commands.
output	Results of events or commands.

input

Used to encapsulate events or commands sent as input to a driver or Identity Manager. All <nds> documents sent as a parameter to Identity Manager or driver interface method should contain exactly one <input>.

Example

See <nds>.

Allowed Content

Element	Description
add	Adds an object when an add event occurs.
modify	Modifies an object when a modify event occurs.
delete	Deletes an object when a delete event occurs.
rename	Renames an object when a rename event occurs.
move	Moves an object when a move event occurs
query	Query command.
query-ex	Query command with a result count limit.
query-schema	Query schema command.
add-association	Adds association command.
modify-association	Modifies an association command.
remove-association	Removes an association command.
init-params	Initialization parameters for a DriverShim, SubscriptionShim, or PublicationShim.
status	Status of the processing of a command or event.
check-password	Checks password against an eDirectory driver object.
modify-password	Modifies an object password when a modify event for a password occurs.
check-object-password	Checks password against an eDirectory object.
sync	Resynchronization or migrate event.
get-named-password	Retrieves a named password for a driver.

Attributes

None

Content Rule

(add | modify | delete | rename | move | query | query-ex | query-schema | add-association | modify-association | remove-association | init-params | status | check-password | modify-password | check-object-password | sync | get-named-password) *

Parent Elements

Element	Description
nds	Top-level element for all Identity Manager and Driver communication.

instance

Used to represent an object in eDirectory or the application as part of the response to a <query> command or a <query-ex> command. <instance> does not necessarily represent the complete state of object, but just the information requested by the <query> or <query-ex>. When returned from a driver, an <association> is required.

Example

```
<instance class-name="User" src-dn="\Users\Samuel">
        <association>1012</association>
        <attr attr-name="Surname">
                <value>Jones</value>
        </attr>
        <attr attr-name="cn">
                <value>Samuel</value>
        </attr>
        <attr attr-name="Given Name">
                <value>Samuel</value>
        </attr>
        <attr attr-name="Telephone Number">
                <value>555-1212</value>
                <value>555-1764</value>
        </attr>
</instance>
```

Allowed Content

Element	Description
association	Unique key of the application object.
parent	The parent or container of an object.
attr	Current state of an attribute.
operation-data	The operation adds additional custom data.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA The name of the base class of the object. The class name is mapped between the application and eDirectory namespaces by the Schema Mapping policy so that Identity Manager sees the name in the eDirectory namespace and a driver sees the name in the application namespace.	#REQUIRED

Attribute	Possible Values	Default Value
event-id	CDATA An identifier used to tag the results of an event or command.	#IMPLIED
qualified-src-dn	CDATA The qualified version of src-dn. Only used for describing objects from eDirectory.	#IMPLIED
src-dn	CDATA The distinguished name of the source object that generated the event in the namespace of the sender.	#IMPLIED
src-entry-id	CDATA The entry ID of the source object that generated the event in the namespace of the sender. Reserved: Should be ignored by the driver.	#IMPLIED

Content Rule

(association ? , parent ? , attr * , operation-data ?)

Parent Elements

Element	Description
output	Results of events or commands.

modify

Used as an event notification from the PublicationShim to Identity Manager that an object is modified in the application. When it is used as a notification, an <association> is required. It is also used as a command from Identity Manager to the SubsciptionShim to modify an object in the application. When it is used as a command, an <association> is required and is the unique key of the object to modify.

Remarks

<add> contains a <modify-attr> for each attribute modified.

A response to <modify> should be a <status> indicating whether or not the <modify> is processed successfully.

Example

```
<modify class-name="User" src-dn="\Sam">
        <association>1012</association>
        <modify-attr attr-name="Given Name">
                <remove-all-values/>
                <add-value>
                        <value>Samuel</value>
                </add-value>
        </modify-attr>
        <modify-attr attr-name="Telephone Number">
                <remove-value>
                        <value>555-1212</value>
                </remove-value>
                <add-value>
                        <value>555-1764</value>
                        <value>555-1765</value>
                </add-value>
        </modify-attr>
</modify>
```

Allowed Content

Element	Description
association	Unique key of the application object.
modify-attr	Modifies an attribute
operation-data	The operation adds additional custom data.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA The name of the base class of the object. The class name is mapped between the application and eDirectory namespaces by the Schema Mapping policy so that Identity Manager sees the name in the eDirectory namespace and a driver sees the name in the application namespace. Required when used as a notification.	#IMPLIED
dest-dn	CDATA The distinguished name of the target object in the namespace of the receiver. Reserved. Should be ignored by the driver.	#IMPLIED
dest-entry-id	CDATA The entry ID of the target object in the namespace of the receiver. Reserved. Should be ignored by the driver.	#IMPLIED
event-id	CDATA An identifier used to tag the results of an event or command.	#IMPLIED
from-merge	true | false True if the command is the result of a merge	false
qualified-src-dn	CDATA The qualified version of src-dn. Only used for describing objects from eDirectory.	#IMPLIED
src-dn	CDATA The distinguished name of the source object that generated the event in the namespace of the sender.	#IMPLIED

Attribute	Possible Values	Default Value
src-entry-id	CDATA The entry id of the source object that generated the event in the namespace of the sender. Reserved. Should be ignored by the driver.	#IMPLIED
timestamp	CDATA Reserved. Should be ignored by the driver.	#IMPLIED

Content Rule

(association ? , modify-attr + , operation-data ?)

Parent Elements

Element	Description
input	Input events or commands.

modify-association

Used to notify Identity Manager that an application object's unique key is modified. <modify-association> should be sent when the unique key is changed for an object that passes the event filter for either the SubscriptionShim or the PublicationShim. <modify-association> can be included in any <output> or <input> from the driver to Identity Manager.

Example

```
<modify-association>
   <association>{BC3E7155-CDF9-d311-9846-0008C76B16C2}</association>
   <association>{CD3F7155-DE09-e311-9846-0008D76C16D2}</association>
</modify-association>
```

Allowed Content

Element	Description
association	Unique key of the application object.
operation-data	The operation adds additional custom data.

Attributes

Attribute	Possible Values	Default Value
event-id	CDATA An identifier used to tag the results of an event or command.	#IMPLIED

Content Rule

(association , association , operation-data ?)

Parent Elements

Element	Description
input	Input events or commands.
output	Results of events or commands.

modify-attr

Used to specify the modified attribute values for a <modify> operation or event.

Remarks

Each <modify-attr> should contain at least one <add-value>, <remove-value>, or <remove-all-values>.

The order of the above elements is significant.

Example

See <modify>.

Allowed Content

Element	Description
remove-value	Removes the specified attribute values.
remove-all-values	Removes all attribute values.
add-value	Adds values.

Attributes

Attribute	Possible Values	Default Value
attr-name	CDATA The name of the attribute. The name is mapped between the application and eDirectory namespaces by the Schema Mapping policy so that Identity Manager sees the name in the eDirectory namespace and a driver sees the name in the application namespace. The Mapping policy uses the class name attribute of the enclosing command or event to determine which class to use for mapping the attribute name.	#REQUIRED
enforce-password-policy	true | false Reserved. Should be ignored by the driver.	#IMPLIED

Content Rule

(remove-value | remove-all-values | add-value) +

Parent Elements

Element	Description
modify	Modifies an object when a modify event occurs.

modify-password

Used as an event notification from the PublicationShim to Identity Manager that an object password is modified in the application. When used as a notification, an <association> is required. Also used as a command from Identity Manager to the SubsciptionShim to modify an object password in the application. When used as a command, an <association> is required and is the unique key of the object to modify.

Remarks

When the target is eDirectory, and <old-password> is specified, the modifyPassword API is used to modify the password. If not specified, the GenerateKeyPair API is used. Using GenerateKeyPair might invalidate authentication credentials for any existing session authenticated as the target object.

When the target is the application, a driver might or might not implement this functionality, depending on the applicability to the application.

A response to <modify-password> should be a <status> indicating whether or not the <modify-password> is processed successfully.

Example

```
<modify-password class-name="User" src-dn="\Sam">
   <association>1012</association>
   <password>mypassword<password>
</modify-password>
```

Allowed Content

Element	Description
association	Unique key of the application object.
old-password	The old authentication password.
password	The authentication password.
operation-data	The operation adds additional custom data.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA The name of the base class of the object. The class name is mapped between the application and eDirectory namespaces by the Schema Mapping policy so that Identity Manager sees the name in the eDirectory namespace and a driver sees the name in the application namespace. Required when used as a notification.	#IMPLIED
dest-dn	CDATA The distinguished name of the target object in the namespace of the receiver.	#IMPLIED
dest-entry-id	CDATA The entry ID of the target object in the namespace of the receiver. Reserved. Should be ignored by the driver.	#IMPLIED
event-id	CDATA An identifier used to tag the results of an event or command.	#IMPLIED
qualified-src-dn	CDATA The qualified version of the src-dn. Only used for describing objects from eDirectory.	#IMPLIED
src-dn	CDATA The distinguished name of the source object that generated the event in the namespace of the sender.	#IMPLIED
src-entry-id	CDATA The entry ID of the source object that generated the event in the namespace of the sender. Reserved. Should be ignored by the driver.	#IMPLIED

Attribute	Possible Values	Default Value
timestamp	CDATA Reserved. Should be ignored by the driver.	#IMPLIED

Content Rule

(association ? , old-password ? , password , operation-data ?)

Parent Elements

Element	Description
input	Input events or commands.

move

Used as an event notification from the PublicationShim to Identity Manager when an object is moved to a different container in the application. When used as a notification, an <association> is required. Also used as a command from Identity Manager to the SubsciptionShim to move an object to a different container in the application. When used as a command, an <association> is required and is the unique key of the object to move.

Remarks

<move> contains a <parent> that specifies the new container. When used as a command, the <parent> can contain an <association>. If it does not contain an association, the driver should not attempt to move the object and should return a <status> level="warning".

A response to <move> should be a <status> indicating whether or not the <move> is processed successfully.

Example

```
<move class-name="User" src-dn="\Users\Samuel" old-src-dn="\Samuel">
   <association>1012</association>
      <parent src-dn="\Users\">
         <association>1013</association>
      </parent>
</move>
```

Allowed Content

Element	Description
association	Unique key of the application object.
parent	The parent or container of an object.
operation-data	The operation adds additional custom data

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA The name of the base class of the object. The class name is mapped between the application and eDirectory namespaces by the Schema Mapping policy so that Identity Manager sees the name in the eDirectory namespace and a driver sees the name in the application namespace.	#IMPLIED

Attribute	Possible Values	Default Value
dest-dn	CDATA	#IMPLIED
	The distinguished name of the target object in the namespace of the receiver.	
	Reserved. Should be ignored by the driver.	
dest-entry-id	CDATA	#IMPLIED
	The entry ID of the target object in the namespace of the receiver.	
	Reserved. Should be ignored by the driver.	
event-id	CDATA	#IMPLIED
	An identifier used to tag the results of an event or command.	
old-src-dn	CDATA	#IMPLIED
	The original distinguished name of the source object that generated the event in the namespace of the sender.	
qualified-old-src-dn	CDATA	#IMPLIED
	The qualified version of the old-src-dn. Only used for describing objects from eDirectory.	
qualified-src-dn	CDATA	#IMPLIED
	The qualified version of the src-dn. Only used for describing objects from eDirectory.	
src-dn	CDATA	#IMPLIED
	The distinguished name of the source object that generated the event in the namespace of the sender.	
	The new distinguished name of the source object that generated the event in the namespace of the sender.	
src-entry-id	CDATA	#IMPLIED
	The entry ID of the source object that generated the event in the namespace of the sender.	
	Reserved. Should be ignored by the driver.	

Attribute	Possible Values	Default Value
timestamp	CDATA Reserved. Should be ignored by the driver.	#IMPLIED

Content Rule

(association ? , parent , operation-data ?)

Parent Elements

Element	Description
input	Input events or commands.

nds

The top-level (document) element of all documents sent as a parameter to or returned from Identity Manager or driver interface method.

Remarks

All <nds> documents sent as a parameter to Identity Manager or driver interface method should contain exactly one <input>.

All <nds> documents returned from Identity Manager or driver interface method should contain exactly one <output>.

Example

```
DirXML sends:

   <nds dtdversion="2.0">
       <source>
          <product version="2.0.0.0">DirXML</product>
          <contact>Novell, Inc.</contact>
       </source>
       <input>
          <modify class-name="User" event-id="0" qualified-src-
dn="\T=ATREE\O=Users\CN=Julia" src-dn="\ATREE\Users\Julia"
    src-entry-id="33967">
             <association state="associated">
                {B43E7155-CDF9-d311-9846-0008C76B16C2}
             </association>
             <modify-attr attr-name="Surname">
                <add-value>
                   <value type="string">Gulia</value>
                </add-value>
             </modify-attr>
          </modify>
       </input>
   </nds>

Shim returns:

   <nds dtdversion="2.0">
      <source>
         <product version="2.0.0.0">Some Application Driver</product>
         <contact>Nobody in particular</contact>
      </source>
      <output>
         <status event-id="0" level="success"/>
      </output>
   </nds>
```

Allowed Content

Element	Description
source	The source or creator of the document.
input	Input events or commands.
output	Results of events or commands.

Attributes

Attribute	Possible Values	Default Value
dtdversion	CDATA Should be set to the major.minor version of Identity Manager that the driver is designed for.	#REQUIRED
ndsversion	CDATA Deprecated as of Identity Manager 2.0.	#IMPLIED

Content Rule

(source ? , (input | output))

Parent Elements

None

nds-name

Used to specify a class or attribute name in the eDirectory namespace.

Example

See <attr-name-map>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
attr-name	Maps an attribute name.
class-name	Maps a class name.

new-name

Used to specify the new name for the object specified by the enclosing <rename> event or command.

Example

See <rename>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
rename	Renames an object when a rename event occurs.

old-password

Used to specify old password for the enclosing <modify-password>.

Example

See <modify-password>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
modify-password	Modifies an object password when a modify event for a password occurs.

operation-data

Used to allow policies to inject an additional custom data payload to be carried along with any event or command. It is stripped from the event or command before it is submitted to the application shim and then reassociated with any corresponding response elements (as determined by matching event-id) after they are returned to Identity Manager.

Remarks

The content of the <operation-data> can be any well-formed XML, but it is recommended that any elements and attributes be placed in a custom namespace to avoid having them confused with standard Identity Manager operations.

The typical use for <operation-data> is to create a policy that supplies additional context on an operation that might be needed by the policy that handles the results of that operation.

For operations whose content is normally PCDATA, there should only be one <operation-data> and it should be after any character data. This is contrary to the content rule specified by the DTD because DTDs for mixed content do not allow more precise specification.

Example

```
<operation-data xmlns:mystuff="http://mystuff.operation.data">
   <mystuff:notify>admin@fred.com</mystuff:notify>
</operation-data>
```

Allowed Content

ANY

Attributes

None

Content Rule

ANY

Parent Elements

Element	Description
add	Adds an object when an add event occurs.
add-association	Adds an association.
check-object-password	Checks the password against an eDirectory object.
check-password	Checks the password against an eDirectory driver object.
delete	Deletes an object when a delete event occurs.
get-named-password	Retrieves a named password for a driver.

Element	Description
init-params	initialization parameters for the DriverShim, SubscriptionShim, or PublicationShim.
instance	Current state of an instance of an object.
modify	Modifies an object when a modify event occurs.
modify-association	Modifies an association command.
modify-password	Modifies an object password when a modify event for a password occurs.
move	Moves an object when a move event occurs.
query	Query command.
query-ex	Query command with a result count limit.
query-schema	Query schema command.
remove-association	Removes an association.
rename	Renames an object when a rename event occurs.
schema-def	Schema definition.
sync	Resynchronization or migrate event.

output

Used to encapsulate events or commands returned to a driver or Identity Manager. All <nds> documents returned from Identity Manager or the driver interface method should contain exactly one <output>.

Example

See <nds>.

Allowed Content

Element	Description
status	Status of the processing of a command or event.
add-association	Adds an association command.
modify-association	Modifies an association command.
remove-association	Removes an association command.
instance	Current state of an instance of an object.
schema-def	Schema definition.
init-params	Initialization parameters for a DriverShim, SubscriptionShim, or PublicationShim.
password	The authentication password.
query-token	Opaque handle for query-ex commands.

Attributes

None

Content Rule

(status | add-association | modify-association | remove-association | instance | schema-def | init-params | password | query-token) *

Parent Elements

Element	Description
nds	Top-level element for all Identity Manager and Driver communication.

parent

Used to specify the destination container for a <move> event or command, or the current container of an object represented by an <instance>.

Remarks

When originating from a driver, the <parent> must contain an <association> containing the unique key of the container object. When originating from Identity Manager, <parent> contains an <association> only if the container object has an established association in eDirectory.

Example

See <move>.

Allowed Content

Element	Description
association	Unique key of the application object.

Attributes

Attribute	Possible Values	Default Value
dest-dn	CDATA The distinguished name of the target object in the namespace of the receiver. Reserved. Should be ignored by the driver.	#IMPLIED
dest-entry-id	CDATA The entry ID of the target object in the namespace of the receiver. Reserved. Should be ignored by the driver.	#IMPLIED
qualified-src-dn	CDATA The qualified version of src-dn. Only used for describing objects from eDirectory.	#IMPLIED
src-dn	CDATA The distinguished name of the source object that generated the event in the namespace of the sender.	#IMPLIED

Attribute	Possible Values	Default Value
src-entry-id	CDATA The entry ID of the source object that generated the event in the namespace of the sender. Reserved. Should be ignored by the driver.	#IMPLIED

Content Rule

(association ?)

Parent Elements

Element	Description
instance	Current state of an instance of an object.
move	Moves an object when a move event occurs.

password

Used to specify the initial password for an object in an <add>, to specify the authentication password for a driver in an <authentication-info> element (it comes from the DirXML-ShimAuthPassword attribute on the DirXML-Driver object), or to return the value of a named password as the result of processing a <get-named-password> command.

Example

See <init-params>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
add	Adds an object when an add event occurs.
authentication-info	Information for connecting and authenticating to the application.
check-object-password	Checks the password against an eDirectory object.
modify-password	Modifies an object password when a modify event for a password occurs.
output	Results of events or commands.

product

Used to specify the product that created the enclosing document.

Example

See <nds>.

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
asn1id	CDATA The ASN.1 Object ID of the product.	#IMPLIED
build	CDATA The build number or timestamp of the product.	#IMPLIED
edition	CDATA The description of edition.	#IMPLIED
instance	CDATA The instance of the product (usually the RDN of the driver object)	#IMPLIED
version	CDATA The version of the product.	#IMPLIED

Content Rule

(#PCDATA)

Parent Elements

Element	Description
source	The source/creator of the document.

publisher-options

Used to specify driver-specific configuration options. It comes from the DirXML-ShimConfigInfo attribute of the DirXML-Driver object in eDirectory. See <driver-config> for details on the contents.

Example

See <driver-config>.

Allowed Content

ANY

Attributes

None

Content Rule

ANY

Parent Elements

Element	Description
driver-config	Driver-specific DriverShim configuration options.
init-params	Initialization parameters for a DriverShim, SubscriptionShim, or PublicationShim.

publisher-state

Used to specify driver-specific state information for the PublicationShim.

Remarks

<publisher-state> can be included inside of <init-params> in any <output> or <input> from the driver to Identity Manager, which instructs Identity Manager to store the contents on the DirXML-DriverStorage attribute of the DirXML-Driver object in eDirectory.

The <publisher-state> stored on the DirXML-Driver object is included in the <init-params> sent to PublicationShim.init() when a driver is started.

Example

See <init-params>.

Allowed Content

ANY

Attributes

None

Content Rule

ANY

Parent Elements

Element	Description
init-params	Initialization parameters for a DriverShim, SubscriptionShim, or PublicationShim.

query

A command used to find and read objects from eDirectory or the application. Full functionality for Identity Manager rules, object migration, etc. depends on full implementation of the query command by the driver.

Remarks

The base object of the <query> is specified by either by the dest-dn attribute or by an <association>. If both are present, the dest-dn attribute takes precedence when querying eDirectory, and the <association> takes precedence when querying a connected application (other than eDirectory). If neither are present, the base object defaults to the root object in a hierarchical namespace or all objects in a non-hierarchical namespace.

Possible scopes for the query are:

Element	Description
entry	The base object only.
subordinates	The objects contained by the base object.
subtree	The objects in the subtree whose root is the base object, including the base object.

By default, all objects in the specified scope are selected. For scopes other than “entry,” selected objects can be further limited by <search-class> and <search-attr>. For scope “entry,” the effect of <search-attr> and <search-class> are undefined.

When there are <search-class> elements, only objects whose base class matches one of the <search-class> elements are selected.

When there are <search-attr> elements, only objects with attributes matching all of the values specified by all of the <search-attr> element is selected.

By default, all object attributes for the selected objects are to be read. The attributes to be read are limited by <read-attr>. To read none of the object attributes, specify a single nameless <read-attr>.

The <parent> of the selected objects is also read if <read-parent> is specified.

A response to <query> should include an <instance> for each of the selected objects.

A response to <query> should also include a <status> indicating whether or not the <query> is processed successfully. It should not be considered an error if no objects exist that match the search criteria.

Example

```
<!-- search the whole application for a User object with the Surname of
Jones -->
<!-- don't read any attributes but read the parent -->
<query class-name="User" event-id="0" scope="subtree">
        <search-class class-name="User"/>
        <search-attr attr-name="Surname">
                <value type="string">Jones</value>
        </search-attr>
        <read-attr/>
        <read-parent/>
</query>
<!-- read the User object whose foreign key is 1011 -->
<!-- read Surname,cn,Given Name and Telephone Number attributes -->
<query class-name="User" event-id="1" scope="entry">
        <association>1011</association>
        <read-attr attr-name="Surname"/>
        <read-attr attr-name="cn"/>
        <read-attr attr-name="Given Name"/>
        <read-attr attr-name="Telephone Number"/>
</query>
```

Allowed Content

Element	Description
association	Unique key of the application object.
search-class	Query search class filter.
search-attr	Query search attribute value filter.
read-attr	Returns the specified object attribute values.
read-parent	Returns the object parent.
operation-data	The operation adds additional custom data.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA The name of the base class of the object. The class name is mapped between the application and eDirectory namespaces by the Schema Mapping policy so that Identity Manager sees the name in the eDirectory namespace and a driver sees the name in the application namespace. This is required for proper schema mapping of any attribute names specified in the search. It should not be used to limit the search.	#IMPLIED
dest-dn	CDATA The distinguished name of the target object in the namespace of the receiver.	#IMPLIED
dest-entry-id	CDATA The entry ID of the target object in the namespace of the receiver. Reserved. Should be ignored by the driver.	#IMPLIED
event-id	CDATA An identifier used to tag the results of an event or command.	#IMPLIED
scope	entry | subordinates | subtree The scope of the query.	subtree

Content Rule

(association ? , (search-class | search-attr | read-attr | read-parent) * , operation-data ?)

Parent Elements

Element	Description
config-object	eDirectory object to use for additional configuration data.
input	Input events or commands.

query-ex

A <query> variant used to limit the number of search results returned at one time. See <query> for general information on searches and queries.

Remarks

The results of a query-ex command might include a <query-token> element. The query-token element is used in subsequent query-ex commands to retrieve additional results. If query-ex results do not contain a query-token element, then all of the available results for that query-ex command are returned. A different query-token might be returned with each result set. It is not sufficient to use only the query-token returned with the first result set.

The query-ex command has additional attributes that the query command does not:

- The max-result-count attribute, which specifies the maximum number of <instance> elements to return as the result of a single query-ex command.
- The cancel attribute, which is used to cancel a query-ex sequence, thereby freeing any resources associated with the search.

A query-ex command sequence might consist of multiple query-ex commands issued sequentially. The first query-ex command establishes the parameters of the search, returns the initial result set, and, if there are more results than can be returned with the initial result set, also returns a
to be used with subsequent query-ex commands. Subsequent query-ex commands contain the query-token element and are issued repeatedly to obtain additional result sets from the initial search. This process continues until no query-token element appears in the result set.

subsequent query-ex commands using a token returned from a previous query-ex do not change the parameters of a search, regardless of any attributes or child elements.

A query-ex sequence can be abandoned before all results are returned by setting the cancel attribute equal to true on a query-ex command.

Not all application shims support query-ex. Those that do report their support to the Metadirectory engine at shim startup time by returning the following as a child of the <instance> element that is returned as the response to the driver identification query:

```
<attr attr-name="query-ex-supported">
   <value type="state">true</value>
</attr>
```

Example

```
<!-- search the whole application for all User objects -->
<!-- don't read any attributes -->
<query-ex class-name="User" event-id="0" scope="subtree" max-result-count="50">
   <search-class class-name="User"/>
   <read-attr/>
</query-ex>

<!-- read additional results based on the above query, assuming that
the results of the above included the query-token illustrated -->
<query-ex>
   <query-
token>rO0ABXNyACxjb20ubm92ZWxsLm5kcy5kaXJ4bWwuZW5naW5lLk5EU1JlYWRlciRUb2tlbuWeJE0g
a5xBAgACSgAFc3RhbXBJAAlzdGF0ZUhhc2h4cAAAAQQLokQbAbjW9w==</query-token>
</query-ex>

<!-- cancel a query-ex before having read all results -->
<query-ex cancel="true">
   <query-
token>rO0ABXNyACxjb20ubm92ZWxsLm5kcy5kaXJ4bWwuZW5naW5lLk5EU1JlYWRlciRUb2tlbuWeJE0g
a5xBAgACSgAFc3RhbXBJAAlzdGF0ZUhhc2h4cAAAAQQLokQbAbjW9w==</query-token>
</query-ex>
```

Allowed Content

Element	Description
association	Unique key of the application object.
query-token	Opaque handle for query-ex commands.
search-class	Query search class filter.
search-attr	Query search attribute value filter
read-attr	Returns the specified object attribute values.
read-parent	Returns the object parent.
operation-data	The operation adds additional custom data.

Attributes

Attribute	Possible Values	Default Value
cancel	true | false If set to true on a query-ex command containing a <query-token> element, then the search is abandoned and all associated resources are freed by the search target.	#IMPLIED

Attribute	Possible Values	Default Value
class-name	CDATA he name of the base class of the object. The class name is mapped between the application and eDirectory namespaces by the Schema Mapping policy so that Identity Manager sees the name in the eDirectory namespace and a driver sees the name in the application namespace. This is required for proper schema mapping of any attribute names specified in the search. It should not be used to limit the search.	#IMPLIED
dest-dn	CDATA The distinguished name of the target object in the namespace of the receiver.	#IMPLIED
dest-entry-id	CDATA The entry ID of the target object in the namespace of the receiver. Reserved. Should be ignored by the driver.	#IMPLIED
event-id	CDATA An identifier used to tag the results of an event or command.	#IMPLIED
max-result-count	CDATA Specifies the maximum number of <instance> elements to return as a result of a query-ex command. Valid values are positive decimal integers.	#IMPLIED
scope	entry | subordinates | subtree The scope of the query. Entry scope makes little sense with query-ex, but is supported.	subtree

Content Rule

(association ? , query-token ? , (search-class | search-attr | read-attr | read-parent) * , operation-data ?)

Parent Elements

Element	Description
input	Input events or commands.

query-schema

A command used read the schema definition from eDirectory or the application. Drivers are not currently required to implement it, but this might change. Drivers are required to implement the DriverShim.getSchema() method, which does exactly the same thing.

Remarks

A response to <query-schema> should be a <schema-def>.

A response to <query> should also include a <status> indicating whether or not the <query> is processed successfully.

Example

```
<query-schema/>
```

Allowed Content

Element	Description
operation-data	The operation adds additional custom data.

Attributes

Attribute	Possible Values	Default Value
event-id	CDATA An identifier used to tag the results of an event or command.	#IMPLIED

Content Rule

(operation-data ?)

Parent Elements

Element	Description
input	Input events or commands.

query-token

Used in conjunction with the query-ex command. The content of the query-token element is an opaque token or handle that is used to refer to unreturned search results that are the result of a query-ex command.

Remarks

A <query-token> element might be returned in the result set of a query-ex command and is used as a child of a <query-ex> element to obtain additional results selected by an initial query-ex command.

Allowed Content

ANY

Attributes

None

Content Rule

ANY

Parent Elements

Element	Description
output	Results of events or commands.
query-ex	Query command with a result count limit.

read-attr

Used to specify the object attributes to be read by the enclosing <query> command.

Example

See <query>.

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
attr-name	CDATA The name of the attribute. The name is mapped between the application and eDirectory namespaces by the Schema Mapping policy so that Identity Manager sees the name in the eDirectory namespace and a driver sees the name in the application namespace. The mapping rule uses the class name attribute of the enclosing command or event to determine which class to use for mapping the attribute name.	#IMPLIED
type	default | xml If type=xml, then the attribute value is parsed as XML and returned as such.	default

Content Declaration

Empty

Parent Elements

Element	Description
query	Query command.
query-ex	Query command with a result count limit.

read-parent

Used to specify that a <parent> should be included in all instances returned by the enclosing <query> command.

Example

See <query>.

Allowed Content

EMPTY

Attributes

None

Content Declaration

Empty

Parent Elements

Element	Description
query	Query command.
query-ex	Query command with a result count limit.

remove-all-values

Used to specify all values that are removed from the attribute specified in the enclosing <modify-attr>.

Example

See <modify>.

Allowed Content

EMPTY

Attributes

None

Content Declaration

Empty

Parent Elements

Element	Description
modify-attr	Modify attribute.

remove-association

Used to notify Identity Manager that a particular unique key is not valid. <remove-association> should generally occur when an invalid association value is sent to the driver by Identity Manager. It can also be useful as part of an Event Transformation policy. Can be included in any <output> or <input> from the driver to Identity Manager.

Example

```
<remove-association>
   {BC3E7155-CDF9-d311-9846-0008C76B16C2}
</remove-association>
```

Allowed Content

#PCDATA

Element	Description
operation-data	The operation adds additional custom data.

Attributes

Attribute	Possible Values	Default Value
event-id	CDATA An identifier used to tag the results of an event or command.	#IMPLIED

Content Rule

(#PCDATA | operation-data) *

Parent Elements

Element	Description
input	Input events or commands.
output	Results of events or commands.

remove-value

Used to specify values removed from the attribute specified in the enclosing <modify-attr>. Each <add-attr> should contain at least one <value>. A driver should gracefully ignore a <remove-value> for a value that does not exist and continue to process the remainder of the enclosing <modify>.

Example

See <modify>.

Allowed Content

Element	Description
value	The attribute value.

Attributes

None

Content Rule

(value +)

Parent Elements

Element	Description
modify-attr	Modify attribute.

rename

Used as an event notification from the PublicationShim to Identity Manager that an object is renamed in the application. When it is used as a notification, an <association> is required. Also used as a command from Identity Manager to the SubsciptionShim to rename an object in the application. When it is used as a command, an <association> is required and is the unique key of the object to rename.

Remarks

<rename> contains a <new-name> that specifies the new name.

A response to <rename> should be a <status> indicating whether or not the <rename> was processed successfully.

Example

```
<rename class-name="User" src-dn="\Samuel" old-src-dn="\Sam">
   <association>1012</association>
   <new-name>Samuel</new-name>
</rename>
```

Allowed Content

Element	Description
association	Unique key of the application object.
new-name	The new name of a renamed object.
operation-data	The operation adds additional custom data.

Attributes

Attribute	Possible Values	Default Value
dest-entry-id	CDATA The entry ID of the target object in the namespace of the receiver. Reserved. Should be ignored by the driver.	#IMPLIED
event-id	CDATA An identifier used to tag the results of an event or command.	#IMPLIED
old-src-dn	CDATA The original distinguished name of the source object that generated the event in the namespace of the sender.	#IMPLIED

Attribute	Possible Values	Default Value
qualified-old-src-dn	CDATA The qualified version of the old-src-dn. Only used for describing objects from eDirectory.	#IMPLIED
qualified-src-dn	CDATA The qualified version of the src-dn. Only used for describing objects from eDirectory.	#IMPLIED
remove-old-name	true | false True if the old name should be removed, false otherwise. Usually only used in X.500 type applications where the name of an object is also an attribute of the object that can exist independently.	true
src-dn	CDATA The distinguished name of the source object that generated the event in the namespace of the sender. The new distinguished name of the source object that generated the event in the namespace of the sender.	#IMPLIED
src-entry-id	CDATA The entry ID of the source object that generated the event in the namespace of the sender. Reserved. Should be ignored by the driver.	#IMPLIED
timestamp	CDATA Reserved. Should be ignored by the driver.	#IMPLIED

Content Rule

(association ? , new-name , operation-data ?)

Parent Elements

Element	Description
input	Input events or commands.

schema-def

Used as part of the response to a query-schema command and to the DriverShim.getSchema() method. It specifies the schema for an application or edirectory. It is also the top-level element of the XML stored in the DirXML-ApplicationSchema attribute of the DirXML-Driver object.

Remarks

The distinguished name format for the application is specified by dn-format or dn-delims (but not both) and must match what the source application actually uses.

One of the standard DN formats should be adequate for most applications, however it might be necessary to specify a custom delimiter set. The eight characters that make up the delimiter set are defined as follows:

- Typed Name Boolean Flag: 0 means names are not typed, 1 means names are typed
- Unicode No-Map Character Boolean Flag: 0 means don't output or interpret unmappable Unicode characters as escaped hex digit strings, such as, \FEFF. The 0xfeff, 0xfffe, 0xfffd, and 0xffff Unicode characters are not accepted by eDirectory.
- Relative RDN Delimiter
- RDN Delimiter
- Name Divider
- Name Value Delimiter
- Wildcard Character
- Escape Character

If RDN Delimiter and Relative RDN Delimiter are the same character, then the orientation of the name is root right, otherwise the orientation is root left.

If there are more than eight characters in the delimiter set, then the extra characters are all considered to be characters that need to be escaped but have no other special meaning within Identity Manager.

Example

```
<schema-def hierarchical="true">
   <class-def class-name="Organization" container="true">
      <attr-def attr-name="Name" case-sensitive="false" multi-valued="false"
naming="true" read-only="false" required="false" type="string"/>
      <attr-def attr-name="Object Path" case-sensitive="false"
multi-valued="false" naming="false" read-only="false" required="true"
type="string"/>
      <attr-def attr-name="Unique Id" case-sensitive="false"
multi-valued="false" naming="false" read-only="false" required="true"
type="string"/>
   </class-def>
   <class-def class-name="Organizational Unit" container="true">
      <attr-def attr-name="Name" case-sensitive="false" multi-valued="false"
naming="true" read-only="false" required="false" type="string"/>
      <attr-def attr-name="Object Path" case-sensitive="false"
multi-valued="false" naming="false" read-only="false" required="true"
type="string"/>
      <attr-def attr-name="Unique Id" case-sensitive="false"
multi-valued="false" naming="false" read-only="false" required="true"
type="string"/>
   </class-def>
   <class-def class-name="User" container="false">
      <attr-def attr-name="cn" case-sensitive="false" multi-valued="false"
```

```
naming="true" read-only="false" required="true" type="string"/>
      <attr-def attr-name="Surname" case-sensitive="false"
multi-valued="false" naming="false" read-only="false" required="false"
type="string"/>
      <attr-def attr-name="Given Name" case-sensitive="false"
multi-valued="false" naming="false" read-only="false" required="false"
type="string"/>
      <attr-def attr-name="Telephone Number" case-sensitive="false"
multi-valued="true" naming="false" read-only="false" required="false"
type="string"/>
      <attr-def attr-name="Object Path" case-sensitive="false"
multi-valued="false" naming="false" read-only="false" required="true"
type="string"/>
      <attr-def attr-name="Unique Id" case-sensitive="false"
multi-valued="false" naming="false" read-only="false" required="true"
type="string"/>
   </class-def>
   <class-def class-name="Bogus" container="false">
      <attr-def attr-name="Whatever" case-sensitive="false"
multi-valued="true" naming="true" read-only="false" required="false"
type="string"/>
      <attr-def attr-name="Object Path" case-sensitive="false"
multi-valued="false" naming="false" read-only="false" required="true"
type="string"/>
      <attr-def attr-name="Unique Id" case-sensitive="false"
multi-valued="false" naming="false" read-only="false" required="true"
type="string"/>
   </class-def>
</schema-def>
```

Allowed Content

Element	Description
class-def	Schema class definition.
operation-data	The operation adds additional custom data.

Attributes

Attribute	Possible Values	Default Value
application-name	CDATA Specifies the name of the application that uses the schema.	#IMPLIED
dn-delims	CDATA The custom DN format used by the application	#IMPLIED
dn-format	dot | qualified-dot | slash | qualified-slash | ldap The DN format used by the application.	slash

Attribute	Possible Values	Default Value
hierarchical	true | false Specifies whether or not the application stores its data in a hierarchical structure.	true

Content Rule

((class-def) * , operation-data ?)

Parent Elements

Element	Description
output	Results of events or commands.

search-attr

Used to specify the object attributes to be searched for by the enclosing <query> command. Each <search-attr> should contain at least one <value>.

Example

See <query>.

Allowed Content

Element	Description
value	The attribute value.

Attributes

Attribute	Possible Values	Default Value
attr-name	CDATA The name of the attribute. The name is mapped between the application and eDirectory namespaces by the Schema Mapping policy so that Identity Manager sees the name in the eDirectory namespace and a driver sees the name in the application namespace. The mapping policies use the class name attribute of the enclosing command or event to determine which class to use for mapping the attribute name	#REQUIRED

Content Rule

(value) +

Parent Elements

Element	Description
query	Query command.
query-ex	Query command with result count limit.

search-class

Used to specify the object base classes to be searched for by the enclosing <query> command.

Example

See <query>.

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA The name of the base class of the object. The class name is mapped between the application and eDirectory namespaces by the Schema Mapping policy so that Identity Manager sees the name in the eDirectory namespace and a driver sees the name in the application namespace.	#REQUIRED

Content Declaration

Empty

Parent Elements

Element	Description
query	Query command.
query-ex	Query command with result count limit.

server

Used to specify the application server for a driver in an <authentication-info> It comes from the DirXML-ShimAuthServer attribute on the DirXML-Driver object.

Example

See <init-params>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
authentication-info	Information for connecting and authenticating to the application.

source

Used to specify the source of the enclosing document.

Example

See <nds>.

Allowed Content

Element	Description
product	The product from which this document originated.
contact	Point of contact for the originating product.

Attributes

None

Content Rule

(product ? , contact ?)

Parent Elements

Element	Description
nds	Top-level element for all Identity Manager and driver communication.

status

Used to return the status of an operation or event. More than one <status> can be returned for each operation or event. <status> can also be included in an input from the driver to Identity Manager when the driver wants to log status of the driver to the Identity Manager log.

Remarks

Possible values for the level attribute are:

Value	Description
success	Operation or event was successful.
warning	Operation or event was partially successful.
error	Operation or event failed.
fatal	A fatal error occurred. The driver should be shut down.
retry	Application server was unavailable. Send this event or operation later.

Standard values for the type attribute include:

Value	Description
app-general	General responses from an application API.
app-authentication	Related to application authentication requests.
app-connection	Identifies a change in the availability of an application.
driver-general	Identifies a response generated by a driver.
driver-status	A driver state change occurred.
password-set-operation	Identifies a response to an application password set or change event.
remoteloader	Messages generated by the Identity Manager Remote Loader.

If a <status> is not returned for a particular event or command, it is assumed to have succeeded.

Example

```
<status event-id="0" level="success"/>
   <status event-id="0" level="warning">Objects in the rear view mirror may appear
closer than they are!</status>
   <status event-id="0" level="warning" type="driver-status">Driver state changed
to Stopped.</status>
```

Allowed Content

ANY

Attributes

Attribute	Possible Values	Default Value
event-id	CDATA An identifier used to tag the results of an event or command. Should be the same as the event-id of the operation or event that this status is associated with	#IMPLIED
level	fatal | error | warning | success | retry The status level.	#REQUIRED
type	CDATA An identifier used to classify the <status> for reporting purposes.	#IMPLIED

Content Rule

ANY

Parent Elements

Element	Description
input	Input events or commands.
output	Results of events or commands.

subscriber-options

Used to specify driver-specific configuration options. It comes from the DirXML-ShimConfigInfo attribute of the DirXML-Driver object in eDirectory. See <driver-config> for details on the contents.

Example

See <driver-config>.

Allowed Content

ANY

Attributes

None

Content Rule

ANY

Parent Elements

Element	Description
driver-config	Driver-specific DriverShim configuration options.
init-params	Initialization parameters for a DriverShim, SubscriptionShim, or PublicationShim.

subscriber-state

Used to specify driver-specific state information for the SubscriptionShim.

Remarks

<subscriber-state> can be included inside of <init-params> in any <output> or <input> from the driver to Identity Manager, which instructs Identity Manager to store the contents on the DirXML-DriverStorage attribute of the DirXML-Driver object in eDirectory.

The <subscriber-state> stored in the DirXML-DriverStorage of the DirXML-Driver object is included in the <init-params> sent to SubscriptionShim.init() when a driver is started.

Example

See <init-params>.

Allowed Content

ANY

Attributes

None

Content Rule

ANY

Parent Elements

Element	Description
init-params	Initialization parameters for a DriverShim, SubscriptionShim, or PublicationShim.

sync

Appears on the Subscriber channel as the result of a resync (manual or automatic) or as the result of a Migrate from Identity Vault operation.

Remarks

The <sync> is generally only visible in the Event Transformation policy. If the Event Transformation policy is used to perform scope-based filtering, then the <sync> element must be taken into account.

The <sync> element does not appear on the Publisher channel unless injected by the application shim or by a policy on the Publisher channel. However, <sync> elements are processed by Publisher channel policies in the case of a Migrate into Identity Vault operation.

Example

```
<sync class-name="User" src-entry-id="3458909" qualified-src-
dn="\T=TREE\O=container\CN=object" src-dn="\TREE\container\object">
   <association>67847262</association>
</sync>
```

Allowed Content

Element	Description
association	Unique key of the application object.
operation-data	The operation adds additional custom data.

Attributes

Attribute	Possible Values	Default Value
cached-time	CDATA The time the event was placed into the driver cache. The format is CCYYMMDDhhmmss.mmmZ, always in UTC.	#IMPLIED
class-id	CDATA Reserved. Should be ignored by the driver.	#IMPLIED
class-name	CDATA The name of the base class of the object. The class name is mapped between the application and eDirectory namespaces by the Schema Mapping policy so that Identity Manager sees the name in the eDirectory namespace and a driver sees the name in the application namespace.	#IMPLIED
from-move	true | false Flag that indicates the sync operation was triggered by an object move in eDirectory.	False

Attribute	Possible Values	Default Value
qualified-src-dn	CDATA	#IMPLIED
	The qualified version of the src-dn. Only used for describing objects from eDirectory.	
src-dn	CDATA	#IMPLIED
	The distinguished name of the source object that generated the event in the namespace of the sender.	
src-entry-id	CDATA	#IMPLIED
	The entry ID of the source object that generated the event in the namespace of the sender.	
timestamp	CDATA	#IMPLIED
	Reserved. Should be ignored by the driver.	

Content Rule

(association ? , operation-data ?)

Parent Elements

Element	Description
input	Input events or commands.

user

Used to specify a user name to authenticate to the application server for a driver in an <authentication-info>. It comes from the DirXML-ShimAuthID attribute on the DirXML-Driver object.

Example

See <init-params>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

#PCDATA

Parent Elements

Element	Description
authentication-info	Information for connecting and authenticating to the application.

value

Used specify an individual attribute value of the attribute specified by the enclosing element.

Remarks

If the type is structured, the <value> contains at least one <component>. If the type is octet, the content of the <value> is base64 encoded binary data. If the type is anything else, the content is text.

The following table shows the types used to represent the various eDirectory syntaxes.

Syntax	Type	Components (Notes)
SYN_UNKNOWN	octet	(Base64-encoded data)
SYN_DIST_NAME	dn	(referential)
SYN_CE_STRING	string	
SYN_CI_STRING	string	
SYN_PR_STRING	string	
SYN_NU_STRING	string	
SYN_CI_LIST	structured	string (1 or more)
SYN_BOOLEAN	state	true or false
SYN_INTEGER	int	
SYN_OCTET_STRING	octet	(Base64-encoded data)
SYN_TEL_NUMBER	teleNumber	
SYN_FAX_NUMBER	structured	faxNumber
		faxBitCount
		faxParameters (Base64-encoded data)
SYN_NET_ADDRESS	structured	netAddrType
		netAddr (Base64-encoded data)
SYN_OCTET_LIST	structured	octet (Base64-encoded data) (1 or more)
SYN_EMAIL_ADDRESS	structured	eMailType
		(1=SMF70,2=SMF71,3=SMTP,4=x400,5=snads, 6=profs,7=Groupwise)
		eMailAddr
SYN_PATH	structured	nameSpace
		volume (referential)
		path

Syntax	Type	Components (Notes)
SYN_REPLICA_POINTER	structured	server (referential)
		replicaType
		replicaNumber
		repeated 0 or more times {
		netAddrType
		netAddr (Base64-encoded data)
		}
SYN_OBJECT_ACL	structured	protectedName
		trustee (referential)
		privileges
SYN_PO_ADDRESS	structured	string (exactly 6)
SYN_TIMESTAMP	structured	seconds
		replicaNumber
		eventId
SYN_CLASS_NAME	classname	
SYN_STREAM	octet	(Base64-encoded data)
SYN_COUNTER	counter	
SYN_BACK_LINK	structured	serverDn (referential)
		remoteId
SYN_TIME	time	
SYN_TYPED_NAME	structured	dn (referential)
		level
		interval
SYN_HOLD	structured	holdEntryDn (referential)
		holdAmount
SYN_INTERVAL	interval	
SYNTAX_COUNT	count	

Example

```
<value type="string">Gulia</value>
<value type="structured">
        <component name="eMailType">3</component>
        <component name="eMailAddr">me@myself.com</component>
</value>
```

Allowed Content

#PCDATA

Element	Description
component	Component of a structured attribute.

Attributes

Attribute	Possible Values	Default Value
association-ref	CDATA The association value (application object unique key) of the object being referenced by this value. This is required on all components that refer to other objects when the value is part of a notification event from the driver. This exists on all values that refer to other objects when the value is part of a command from Identity Manager if the referenced object has an established association in eDirectory. If a <component> is referential, the association-ref is on the <component> rather than the <value>.	#IMPLIED
naming	true | false Reserved. Should be ignored by the driver.	false
timestamp	CDATA Reserved. Should be ignored by the driver.	#IMPLIED
type	string | teleNumber | int | state | counter | dn | interval | octet | time | structured The data type of the value.	#IMPLIED

Content Rule

(#PCDATA | component) *

Parent Elements

Element	Description
add-attr	Adds an attribute.
add-value	Adds values.
attr	Matches an attribute.
match-attr	Inputs events or commands. Deprecated as of Identity Manager 2.0.
remove-value	Removes specified attribute values.
required-attr	Required attribute. Deprecated as of Identity Manager 2.0.
search-attr	Query search attribute value filter.

3.2 Deprecated NDS DTD Elements

The following NDS DTD elements have been deprecated as of Identity Manager 2.0.

copy-attr

Deprecated as of Identity Manager 2.0. Use DirXMLScript <token-op-attr> instead.

Remarks

<copy-attr> is used as a token to specify a string replacement in the distinguished name generated by the enclosing <placement>.

The replacement string is generated by copying the first value for the attribute specified by attr-name from the <add> event that is being processed. If the attribute does not exist, then the enclosing <placement-rule> is skipped. Structured attribute types are not supported.

Example

See <placement>.

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
attr-name	CDATA The name of the attribute. The name is mapped between the application and eDirectory namespaces by the Schema Mapping policy so that Identity Manager sees the name in the eDirectory namespace and a driver sees the name in the application namespace. The mapping rule uses the class name attribute of the enclosing command or event to determine which class to use for mapping the attribute name.	#REQUIRED

Content Declaration

Empty

Parent Elements

Element	Description
NDS DTD	Object placement specifier. Deprecated as of Identity Manager 2.0.

copy-name

Deprecated as of Identity Manager 2.0. Use DirXMLScript <token-src-dn> instead.

Remarks

<copy-name> is used as a token to specify a string replacement in the distinguished name generated by the enclosing <placement>.

The replacement string is generated by copying the unqualified portion of the leaf-most component of the src-dn attribute from the <add> event that is being processed. If the src-dn does not exist, then the enclosing <placement-rule> is skipped.

Example

See <placement-rules>.

Allowed Content

EMPTY

Attributes

None

Content Declaration

Empty

Parent Elements

Element	Description
placement	Object placement specifier. Deprecated as of Identity Manager 2.0.

copy-path

Deprecated as of Identity Manager 2.0. Use DirXMLScript <token-src-dn> instead.

Remarks

<copy-path> is used as a token to specify a string replacement in the distinguished name generated by the enclosing <placement>.

The replacement string is generated by copying the src-dn attribute from the <add> event that is being processed. A conversion from the src-dn-format to the dest-dn-format of the enclosing <placement-rules> is performed if the formats are different. Conversion from a typeless (unqualified) format to a typed (qualified) format is unsupported unless the source is eDirectory.

Example

See <placement-rules>.

Allowed Content

EMPTY

Attributes

None

Content Declaration

Empty

Parent Elements

Element	Description
placement	Object placement specifier. Deprecated as of Identity Manager 2.0.

copy-path-suffix

Deprecated as of Identity Manager 2.0. Use DirXMLScript <token-unmatched-src-dn> instead.

Remarks

<copy-path-suffix> is used as a token to specify a string replacement in the distinguished name generated by the enclosing <placement>.

The replacement string is generated by copying src-dn attribute from the <add> event that is being processed, and then stripping away the portion of the src-dn matched by a <match-path> in the enclosing <placement-rule>. If no <match-path> was specified then the whole src-dn is copied. A conversion from the src-dn-format to the dest-dn-format of the enclosing <placement-rules> is performed if the formats are different. Conversion from a typeless (unqualified) format to a typed (qualified) format is unsupported unless the source is eDirectory.

Example

See <placement-rules>.

Allowed Content

EMPTY

Attributes

None

Content Declaration

Empty

Parent Elements

Element	Description
placement	Object placement specifier. Deprecated as of Identity Manager 2.0.

create-rule

Deprecated as of Identity Manager 2.0. Use DirXMLScript <rule> instead.

Remarks

<create-rule> is used to specify the criteria for creating a new object as a result of an <add> event.

When a <create-rule> is evaluated, it first checks whether or not this is a suitable rule for the <add> event in question. It does this by checking if a class name is specified by the rule. If so, the rule is only suitable if the class name matches the class name on the event. It then checks if any <match-attr> criteria are specified by the rule. If so, the rule is only suitable if the <add> contains all the attribute values required by the <match-attr>. If a rule is determined to not be suitable, it is skipped.

When a suitable rule is found, the <match-class> is evaluated to see if it has a value for all of the <required-attr> that do not contain a default value. If not, the <add> is vetoed; otherwise, it is allowed. Then any required attributes with default values that were missing from the <add> are filled in. If the write-back attribute of the <required-attr> element is set, the missing values are also written back to the source object. The template-dn attribute is filled in if a <template> is specified.

Example

See <create-rules>.

Allowed Content

Element	Description
match-attr	Matches an attribute. Deprecated as of Identity Manager 2.0.
read-attr	Required attribute. Deprecated as of Identity Manager 2.0.
template	Specifies a template. Deprecated as of Identity Manager 2.0.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA The name of the base class of the objects this rule applies to in the eDirectory namespace. If empty or not present, then this rule applies to all base classes	#IMPLIED
description	CDATA Description of this rule, primarily for use in ConsoleOne.	#IMPLIED

Content Rule

(match-attr * , required-attr * , template ?)

Parent Elements

Element	Description
create-rules	Top-level element for object creation rules. Deprecated as of Identity Manager 2.0.

create-rules

Deprecated as of Identity Manager 2.0. Use DirXMLScript <policy> instead.

Remarks

<create-rules> is the top-level (document) element for object creation rules. Object creation rules are stored in the DirXML-XmlData attribute of a DirXML-Rule object that is pointed to by the DirXML-CreateRule attribute of a DirXML-Subscriber or DirXML-Publisher object.

In the Subscriber channel, the source is eDirectory, and the destination is the application. In the Publisher channel, the source is the application and the destination is eDirectory.

Object creation rules are used to determine whether or not to create a new object in the destination as a result of an <add> event in the source. (Identity Manager automatically converts <modify> into <add> for events from unassociated objects). Object creation rules are applied only after any existing Matching rules are applied and fail to find a matching object in the destination.

<create-rules> contains 0 or more <create-rule> elements. The creation rule processor evaluates each <create-rule> in order until a suitable rule is found. That rule then vetoes or allows the object creation and fills in any default attributes and templates specified. If no suitable <create-rule> is found, then the object creation is allowed.

Example

```
<create-rules>
        <!-- For all Users in the Defense organization require Given Name-->
        <!-- Surname, and Security Clearance. Create using the -->
        <!-- templates\Secure User template -->
        <create-rule class-name="User">
                <match-attr attr-name="OU">
                        <value>Defense</value>
                </match-attr>
                <required-attr attr-name="Given Name"/>
                <required-attr attr-name="Surname"/>
                <required-attr attr-name="Security Clearance"/>
                <template template-dn="templates\Secure User"/>
        </create-rule>
        <!-- For all other Users require Given Name and Surname. -->
        <!-- Default the value of Security Clearance to None -->
        <!-- Don't use a template for creation -->
        <create-rule class-name="User">
                <required-attr attr-name="Given Name"/>
                <required-attr attr-name="Surname"/>
                <required-attr attr-name="Security Clearance">
                        <value>None</value>
                </required-attr>
        </create-rule>
</create-rules>
```

Allowed Content

Element	Description
create-rule	Object creation rule. Deprecated as of Identity Manager 2.0.

Attributes

None

Content Rule

(create-rule) *

Parent Elements

None

match-attr

Deprecated as of Identity Manager 2.0. Use DirXMLScript <if-op-attr> for an object creation and Placement policy or <do-find-matching-object> and <arg-match-attr> for an object Matching policy.

Remarks

<match-attr> is used to specify:

- Rule selection criteria for the enclosing <create-rule> or <placement-rule>. When used as such, it must contain at least one <value>.
- Object selection criteria for the enclosing <matching-rule>. When used as such, it must not contain a <value>.

Example

See <create-rules>.

Allowed Content

Element	Description
value	The attribute value.

Attributes

Attribute	Possible Values	Default Value
attr-name	CDATA The name of the required attribute in the eDirectory namespace.	#REQUIRED

Content Rule

(value) *

Parent Elements

Element	Description
create-rule	Object creation rule. Deprecated as of Identity Manager 2.0.
matching-rule	Object matching rule. Deprecated as of Identity Manager 2.0.
placement-rule	Object placement rule. Deprecated as of Identity Manager 2.0.

match-class

Deprecated as of Identity Manager 2.0. Use DirXMLScript <if-object-class> instead.

Remarks

<match-class> is used to specify rule selection criteria for the enclosing <matching-rule> or <placement-rule>.

Example

See <matching-rules> and <placement-rules>.

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA The name of the base class in the eDirectory namespace.	#REQUIRED

Content Declaration

Empty

Parent Elements

Element	Description
matching-rule	Object matching rule. Deprecated as of Identity Manager 2.0.
placement-rule	Object matching rule. Deprecated as of Identity Manager 2.0.

match-path

Deprecated as of Identity Manager 2.0. Use DirXMLScript <if-src-dn> for object placement policy or <do-find-matching-object> and <arg-dn> for object matching policy.

Remarks

<match-path> is used to specify:

- Rule selection criteria for the enclosing <placement-rule>. When used as such, the src-dn attribute of the source <add> event is compared with prefix and is considered a match if the src-dn is in the subtree whose root is prefix. The namespace of the path is the same as the event source.
- Object selection criteria for the enclosing <matching-rule>. When used as such, prefix is used as the dest-dn for the <query> generated by the enclosing rule. The namespace of the path is the same as the event destination.

When the namespace of the path is eDirectory, the format is slash format, for example, \treename\container\...\leaf. If the leading \ is omitted, the path is assumed to be relative to the tree root.

When the namespace of the path is the application namespace, the format of the path is application dependent and should be documented by the driver writer.

Example

See <matching-rules> and <placement-rules>.

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
prefix	CDATA The root-most portion of the path or distinguished name to match.	#REQUIRED

Content Declaration

EMPTY

Parent Elements

Element	Description
matching-rule	Object matching rule. Deprecated as of Identity Manager 2.0.
placement-rule	Object matching rule. Deprecated as of Identity Manager 2.0.

matching-rule

Deprecated as of Identity Manager 2.0. Use DirXMLScript <rule> and <do-find-matching-object> instead.

Remarks

<matching-rule> is used to specify the criteria for finding a matching object for automatic association of a new object as a result of an <add> event.

When a <matching-rule> is evaluated, it first checks whether or not this is a suitable rule for the <add> event in question. It does this by checking if any <match-class> elements are specified by the rule. If so then the rule is only suitable if the class name on the event matches the class name on one of the <match-class> elements. It then checks if any <modify-attr> criteria are specified by the rule. If so, the rule is only suitable if the <add> contains an attribute value for each attribute specified by a <match-attr>. If a rule is determined to not be suitable, it is skipped.

When a suitable rule is found, a <query> is generated based on the criteria specified by the rule (<match-attr and <match-path>), and the class name and attribute values specified by the <add>. This query is sent to the destination (eDirectory or driver). Any <instance> elements returned are considered matches.

Example

See <matching-rules>.

Allowed Content

Element	Description
match-class	Matches a class name. Deprecated as of Identity Manager 2.0.
match-path	Matches a path. Deprecated as of Identity Manager 2.0.
match-attr	Matches an attribute. Deprecated as of Identity Manager 2.0.

Attributes

Attribute	Possible Values	Default Value
description	CDATA Description of this rule, primarily for use in ConsoleOne.	#IMPLIED

Content Rule

(match-class * , match-path ? , match-attr *)

Parent Elements

Element	Description
matching-rules	Top-level element for object matching rules. Deprecated as of Identity Manager 2.0.

matching-rules

Deprecated as of Identity Manager 2.0 - use DirXMLScript <policy>.

Remarks

<matching-rules> is the top level (document) element for object matching rules. Object matching rules are stored in the DirXML-XmlData attribute of a DirXML-Rule object that is pointed to by the DirXML-MatchingRule attribute of a DirXML-Subscriber or DirXML-Publisher object.

In the Subscriber channel, the source is eDirectory, and the destination is the application. In the Publisher channel the source is the application and the destination is eDirectory.

Object matching rules are used to try to find a matching object in the destination for an unassociated object in the source as a result of an <add> event in the source. (Note that DirXML automatically converts <modify> into <add> for events from unassociated objects). Object matching rules are applied before deciding if a new object should be created in the destination.

<matching-rules> contains 0 or more <matching-rule> elements. The matching rule processor evaluates each <matching-rule> in order until one or more matching objects in the destination are found.

If exactly one matching object is found, that object is automatically associated with the source object and Identity Manager attempts to reconcile any differences in the attribute values of the two objects as allowed by the Publisher and Subscriber filters.

If more than one matching object is found, an error is signaled and the object either has to be manually associated or the object matching rules has to be modified to be more specific.

If no matching objects are found, Identity Manager continues processing the event.

Example

```
<matching-rules>
        <!-- for Users, first try to match on Surname, Given Name and
Location -->
        <matching-rule>
                <match-class class-name="User"/>
                <match-attr attr-name="Surname"/>
                <match-attr attr-name="Given Name"/>
                <match-attr attr-name="Location"/>
        </matching-rule>
        <!-- for Users, then try to match on Surname only in -->
        <!-- the o=novell subtree -->
        <matching-rule>
                <match-class class-name="User"/>
                <match-path prefix="o=novell"/>
                <match-attr attr-name="Surname"/>
        </matching-rule>
        <!-- for all classes try to match on CN only -->
        <matching-rule>
                <match-attr attr-name="CN"/>
        </matching-rule>
</matching-rules>
```

Allowed Content

Element	Description
matching-rule	Object matching rule. Deprecated as of Identity Manager 2.0.

Attributes

None

Content Rule

(matching-rule *)

Parent Elements

None

placement

Deprecated as of Identity Manager 2.0. Use DirXMLScript <do-set-op-dest-dn> instead.

Remarks

<placement> is used to specify the distinguished name for an object that is about to be created and match the selection criteria for the enclosing <placement-rule>.

The DN is generated by concatenating in order the text and the text substitutions generated by the enclosed token elements. Any leading or trailing white space is removed unless it was enclosed by a CDATA section.

Example

See <placement-rules>.

Allowed Content

#PCDATA

Element	Description
copy-name	Copies a name token. Deprecated as of Identity Manager 2.0.
copy-attr	Copies an attribute token. Deprecated as of Identity Manager 2.0.
copy-path	Copies a path token. Deprecated as of Identity Manager 2.0.
copy-path-suffix	Copies a path token. Deprecated as of Identity Manager 2.0.

Attributes

None

Content Rule

(#PCDATA | copy-name | copy-attr | copy-path | copy-path-suffix) *

Parent Elements

Element	Description
placement-rule	Object placement rule. Deprecated as of Identity Manager 2.0.

placement-rule

Deprecated as of Identity Manager 2.0. Use DirXMLScript <rule>.

Remark

<placement-rule> is used to specify the criteria for generating a distinguished name for an object that is about to be created.

When a <placement-rule> is evaluated, it first checks whether or not this is a suitable rule for the <add> event in question. It does this by checking if any <match-class> elements are specified by the rule. If so, the rule is only suitable if the class-name on the event matches the class-name on one of the <match-class> elements. It then checks if any <match-attr> criteria are specified by the rule. If so, the rule is only suitable if the <add> contains all the attribute values required by the <match-attr>. It then checks if any <match-path> criteria are specified by the rule. If so, the src-dn of the <add> must be in at least one of the subtrees specified by a <match-path>. If a rule is determined to not be suitable, it is skipped.

When a suitable rule is found, the <placement> is evaluated to generate a value for the dest-dn attribute on the <add>.

If no suitable rule is found, the dest-dn is left blank and processing of the <add> continues.

Example

See <placement-rules>.

Allowed Content

Element	Description
match-class	Matches a class name. Deprecated as of Identity Manager 2.0.
match-path	Matches a path. Deprecated as of Identity Manager 2.0.
match-attr	Matches an attribute. Deprecated as of Identity Manager 2.0.
placement	Object placement specifier. Deprecated as of Identity Manager 2.0.

Attributes

Attribute	Possible Values	Default Value
description	CDATA Description of this rule, primarily for use in ConsoleOne.	#IMPLIED

Content Rule

(match-class * , match-path * , match-attr * , placement)

Parent Elements

Element	Description
placement-rules	Top-level element for object placement rules. Deprecated as of Identity Manager 2.0.

placement-rules

Deprecated as of Identity Manager 2.0. Use DirXMLScript <policy> instead.

Remarks

<placement-rules> is the top level (document) element for object placement rules. Object matching rules are stored in the DirXML-XmlData attribute of a DirXML-Rule object that is pointed to by the DirXML-PlacementRule attribute of a DirXML-Subscriber or DirXML-Publisher object.

In the Subscriber channel, the source is eDirectory, and the destination is the application. In the Publisher channel, the source is the application and the destination is eDirectory.

Object placement rules are used to generate a distinguished name for an object that is about to be created as the result of an <add> event in the source. (Identity Manager automatically converts <modify> into <add> for events from unassociated objects). Object placement rules are applied only after any existing creation rules are applied and the <add> is not vetoed.

<placement-rules> contains 0 or more <placement-rule> elements. The placements rule processor evaluates each <placement-rule> in order until a suitable rule is found. That rule then fills in the dest-dn attribute of the <add>.

The distinguished name format for the source event is specified by src-dn-format or src-dn-delims (but not both) and must be match what the source application actually uses. The distinguished name format for the destination event is specified by dest-dn-format or dest-dn-delims (but not both) and must match what the destination application actually uses. The format used by eDirectory through Identity Manager is slash.

One of the standard DN formats should be adequate for most applications, but it might be necessary to specify a custom delimiter set. The eight characters that make up the delimiter set are defined as follows:

- Typed Name Boolean Flag: 0 means names are not typed, 1 means names are typed
- Unicode* No-Map Character Boolean Flag: 0 means don't output or interpret unmappable Unicode characters as escaped hex digit strings, such as \FEFF. The Unicode characters 0xfeff, 0xfffe, 0xfffd, and 0xffff are not accepted by eDirectory.
- Relative RDN Delimiter
- RDN Delimiter
- Name Divider
- Name Value Delimiter
- Wildcard Character
- Escape Character

If the RDN Delimiter and Relative RDN Delimiter are the same character, then the orientation of the name is root right, otherwise the orientation is root left.

Example

```
<placement-rules src-dn-format="slash" dest-dn-format="ldap">
   <!-- for Users coming from the subtree \Tree\novell in eDirectory
-->
   <!-- place them in the same relative hierarchy under o=novell -->
        <placement-rule>
                <match-class class-name="User"/>
                <match-path prefix="\TREE\novell"/>

<placement><copy-path-suffix/>,o=novell</placement>
        </placement-rule>
        <!-- for all other users and groups -->
        <!-- place them in the department container under novell -->
        <placement-rule>
                <match-class class-name="User"/>
                <match-class class-name="Group"/>
               <placement>cn=<copy-name/>,ou=<copy-attr attr-name="OU"/>,o=novell</
placement>
        </placement-rule>
        <!-- for everything else, try to mirror the hierarchy -->
        <placement-rule>
                <placement><copy-path/></placement>
        </placement-rule>
</placement-rules>
```

Allowed Content

Element	Description
placement-rule	Object placement rule. Deprecated as of Identity Manager 2.0.

Attributes

Attribute	Possible Values	Default Value
dest-dn-delims	CDATA Custom delimiters for the dest-dn.	#IMPLIED
dest-dn-format	dot | qualified-dot | slash | qualified-slash | ldap Format of the src-dn.	slash
src-dn-delims	CDATA Custom delimiters for the src-dn.	#IMPLIED
src-dn-format	dot | qualified-dot | slash | qualified-slash | ldap Format of the src-dn.	slash

Content Rule

(placement-rule *)

Parent Elements

None

required-attr

Deprecated as of Identity Manager 2.0. Use DirXMLScript <do-veto-if-op-attr-not-available> or <do-set-default-attr-value> instead.

Remarks

<required-attr> is used to specify an attribute required to create an object as part of the criteria for the enclosing <create-rule>.

<required-attr> can contain one or more <value> elements. If it contains any value elements, there are used as default values if the <add> event did not specify that attribute. If no default values are specified, then the <add> event is vetoed unless it contains an <add-attr> corresponding to the attr-name of the <required-attr>.

Example

See <create-rules>.

Allowed Content

Element	Description
value	The attribute value.

Attributes

Attribute	Possible Values	Default Value
attr-name	CDATA The name of the required attribute in the eDirectory namespace.	#REQUIRED
write-back	true | false Set to true if the default value should also be written back to the source object.	false

Content Rule

(value) *

Parent Elements

Element	Description
create-rule	Object creation rule. Deprecated as of Identity Manager 2.0.

template

Deprecated as of Identity Manager 2.0. Use DirXMLScript <do-set-op-template-dn> instead. Used to specify a template that is used for object creation as part of the criteria specified by the enclosing <create-rule>.

Example

See <create-rules>.

Allowed Content

EMPTY

Attributes

Attribute	Possible Values
template-dn	CDATA The DN of the template that is used in the destination namespace.

Content Rule

Empty

Parent Elements

Element	Description
create-rule	Object creation rule. Deprecated as of Identity Manager 2.0.

4 Map DTD

An Identity Manager mapping table is used by a DirXML Script policy to map a set of values to another set of corresponding values.

An Identity Manager mapping table consists of a top level <mapping-table> that contains a set of <col-def> elements and a set of <row> elements. A <col-def> defines the name of each column and the type that it contains. A <row> consists of a set of <col> elements. DirXML Script uses <token-map> to map a value using a specified key column to a different value or values in a specified value column. A given key column maps to multiple rows and therefore multiple values from the value column can be specified.

See "Map DTD Elements" on page 155 for a list of all of the elements in the Map DTD.

4.1 Map DTD Elements

Element	Description
col	Mapping table column within a row.
col-def	Column definition.
mapping-table	Mapping table.
row	Mapping table row.

col

Defines the value of a column within a row in the mapping table.

Example

See <mapping-table>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
row	Mapping table row.

col-def

Defines a column in the mapping table. The name of the column is specified by name. The type of the column is specified by type. The type of the column defaults to nocase and is used; to determine the comparison rules used when the column is used a the key column for a mapping operation.

Remarks

The legal values for type are:

Value	Description
case	Character-by-character case sensitive comparison.
nocase	Character-by-character case insensitive comparison
numeric	Numeric comparison.

Example

See <mapping-table>.

Allowed Content

EMPTY

Attributes

Attribute	Values	Default Value
name	CDATA	#REQUIRED
	Name of the column	
type	case | nocase | numeric	nocase
	type of the column	

Content Declaration

Empty

Parent Elements

Element	Description
mapping-table	Mapping table.

mapping-table

A mapping table consists of a top level <mapping-table> that contains a set of <col-def> elements and a set of <row> elements. A <col-def> defines the name of each column and the type that it contains. A <row> consists of a set of <col> elements. DirXML Script uses <token-map> to map a value using a specified key column to a different value or values in a specified value column. A given key column map to multiple rows and therefore multiple values from the value column can be specified.

Example

```
<mapping-table>
   <col-def name="dept" type="nocase"/>
   <col-def name="code" type="nocase"/>
   <col-def name="location" type="nocase"/>
   <row>
      <col>Engineering</col>
      <col>00001</col>
      <col>New York</col>
   </row>
   <row>
      <col>Sales</col>
      <col>00002</col>
      <col>London</col>
   </row>
   <row>
      <col>Accounting</col>
      <col>00003</col>
      <col>Paris</col>
   </row>
   <row>
      <col>Marketing</col>
      <col>00004</col>
      <col>Rome</col>
   </row>
</mapping-table>
```

Allowed Content

Element	Description
col-def	Column definition.
row	Mapping table row.

Attributes

None

Content Rule

(col-def * , row *)

Parent Elements

None

row

A <row> defines a row in the mapping table. The values for the columns within the row are defined by the enclosing <col> elements and correspond to the columns defined for the mapping table and must occur in the same order as the <col-def> elements. If there are fewer columns than there are columns defined for the table, then the missing columns will be assumed to be blank. If there are more columns in the row than are defined in the table, the additional columns are ignored.

Example

See <mapping-table>.

Allowed Content

Element	Description
col	Mapping table column within a row.

Attributes

None

Content Rule

(col *)

Parent Elements

Element	Description
mapping-table	Mapping table.

5 DirXML Script DTD

DirXML Script is the primary method of implementing policies in the Novell Identity Manager Metadirectory engine. DirXML Script describes a <policy> that is implemented by an ordered set of <rule> elements. A <rule> consists of a set of <conditions> to be tested and an ordered set of <actions> to be performed when the <conditions> are met.

See "DirXML Script DTD Elements" on page 161 for a list of all of the elements in the DirXML Script DTD.

5.1 DirXML Script DTD Elements

Element	Description
actions	Actions that are performed by a <rule>.
and	Logical conjunction.
arg-actions	Actions argument.
arg-association	Association argument.
arg-component	Component argument.
arg-conditions	Conditions argument.
arg-dn	DN argument.
arg-match-attr	Match attribute argument.
arg-node-set	Node set argument.
arg-object	Java* Object argument
arg-password	Password argument.
arg-string	String argument
arg-value	Value argument.
comment	Long description of a rule.
component	Value component.
conditions	Conditions under which the actions of a rule are performed.
description	Description of a policy or a rule.
do-add-association	Associates the current object.
do-add-dest-attr-value	Adds a value to an attribute in the destination data store.
do-add-dest-object	Adds an object in the destination data store.

Element	Description
do-add-role	Adds a role assignment to a specified object.
do-add-resource	Adds a resource assignment to a specified object.
do-add-src-attr-value	Adds a value to an attribute in the source data store.
do-add-src-object	Adds an object in the source data store.
do-append-xml-element	Appends a custom XML element to existing elements.
do-append-xml-text	Appends custom XML text to existing elements.
do-break	Stops processing the current operation with this policy.
do-clear-dest-attr-value	Clears all values of an attribute in the destination data store.
do-clear-op-property	Clears an operation property.
do-clear-src-attr-value	Clears all values of an attribute in the source data store.
do-clear-sso-credential	Clears a credential in an SSO credential store.
do-clone-op-attr	Applies all operations on an attribute in the current operation to a different attribute.
do-clone-xpath	Clones and appends a set of nodes to existing elements.
do-delete-dest-object	Deletes an object in the destination data store.
do-delete-src-object	Deletes an object in the source data store.
do-find-matching-object	Automatically associates the current object.
do-for-each	Repeats actions for each node in a node set.
do-generate-event	Generates a user-defined event.
do-if	Conditionally perform actions.
do-implement-entitlement	Implements an entitlement.
do-move-dest-object	Moves an object in the destination data store.
do-move-src-object	Moves an object in the source data store.
do-reformat-op-attr	Changes the format of all values of a particular attribute in the current operation.
do-remove-association	Disassociates an application object.
do-remove-dest-attr-value	Removes a value from an attribute in the destination data store.
do-remove-role	Removes a role assignment from a specified object.
do-remove-resource	Removes a resource assignment from a specified object.
do-remove-src-attr-value	Removes a value from an attribute in the source data store.
do-rename-dest-object	Renames an object in the destination data store.
do-rename-op-attr	Changes an attribute name for all operations on that attribute in the current operation.

Element	Description
do-rename-src-object	Renames an object in the source data store.
do-send-email	Generates an e-mail notification.
do-send-email-from-template	Generates an e-mail notification using SMTP configuration and e-mail template objects.
do-set-default-attr-value	Sets the default value for an attribute created in the destination data store
do-set-dest-attr-value	Sets the value of an attribute in the destination data store.
do-set-dest-password	Sets the password for the current object in the destination data store.
do-set-local-variable	Sets the value of a local variable.
do-set-op-association	Sets that association value for the current operation.
do-set-op-class-name	Sets the object class name for the current operation.
do-set-op-dest-dn	Sets the destination DN for the current operation.
do-set-op-property	Sets an operation property.
do-set-op-src-dn	Sets the source DN for the current operation.
do-set-op-template-dn	Sets the template DN for the current add operation.
do-set-src-attr-value	Sets the value of an attribute in the source data store.
do-set-src-password	Sets the password for the current object in the source data store.
do-set-sso-credential	Sets a credential in an SSO credential store.
do-set-sso-passphrase	Sets a passphrase in an SSO credential store.
do-set-xml-attr	Sets custom XML attributes on existing elements.
do-start-workflow	Starts a workflow.
do-status	Reports status.
do-strip-op-attr	Strips an attribute from the current operation.
do-strip-xpath	Strips arbitrary data from the current operation and from any nodeset-type variable.
do-trace-message	Sends a trace message.
do-veto	Vetoes the current operation.
do-veto-if-op-attr-not-available	Vetoes the current operation if a particular attribute is not available in the operation.
do-while	Repeat actions while a condition is true.
if-association	Tests an association.
if-attr	Tests an attribute in the current operation or the current object in the source data store.
if-class-name	Tests the object class of the current operation.

Element	Description
if-dest-attr	Tests an attribute of the current object in the destination data store.
if-dest-dn	Tests the destination DN of the current operation.
if-entitlement	Tests an entitlement of the current object.
if-global-variable	Tests a global variable.
if-local-variable	Tests a local variable.
if-named-password	Tests a named password.
if-op-attr	Tests an attribute in the current operation.
if-op-property	Tests an operation property.
if-operation	Tests the name of the current operation.
if-password	Tests the password of the current operation.
if-src-attr	Tests an attribute of current object in the source data store.
if-src-dn	Tests the source DN of the current operation.
if-xml-attr	Tests an XML attribute of the current operation.
if-xpath	Tests an XPath expression.
include	Includes rules from another policy.
or	Logical disjunction.
policy	A policy.
rule	Rules within a policy.
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.

Element	Description
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.

Element	Description
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

actions

The <actions> that are performed when <conditions> of the enclosing <rule> are met. All individual <actions> are represented by an element of the form <do-*>.

Remarks

Most actions take arguments that further describe the action to be taken. Arguments that take a fixed string that never changes at runtime are represented by attributes on the action element. Arguments that can be re-evaluated at runtime are represented by child elements of the form <arg-*>. The content of most of these arguments consists of a set of tokens represented by elements of the form <token-*> (exceptions are noted on the documentation for the individual arguments). The individual tokens are expanded at runtime based on the rule evaluation context and the results of the expansion and are concatenated together to form the actual argument.

Example

See <policy>.

Allowed Content

Element	Description
do-add-association	Associates the current object.
do-add-dest-attr-value	Adds a value to an attribute in the destination data store.
do-add-dest-object	Adds an object in the destination data store.
do-add-src-attr-value	Adds a value to an attribute in the source data store.
do-add-src-object	Adds an object in the source data store.
do-append-xml-element	Appends a custom XML element to existing elements.
do-append-xml-text	Appends custom XML text to existing elements.
do-break	Stops processing the current operation with this policy.
do-clear-dest-attr-value	Clears all values of an attribute in the destination data store.
do-clear-op-property	Clears an operation property.
do-clear-src-attr-value	Clears all values of an attribute in the source data store.
do-clear-sso-credential	Clears a credential in an SSO credential store.
do-clone-op-attr	Applies all operations on an attribute in the current operation to a different attribute.
do-clone-xpath	Clones and appends a set of nodes to existing elements.
do-delete-dest-object	Deletes an object in the destination data store.

Element	Description
do-delete-src-object	Deletes an object in the source data store.
do-find-matching-object	Automatically associates the current object.
do-for-each	Repeats actions for each node in a node set.
do-generate-event	Generates a user-defined event.
do-if	Conditionally perform actions.
do-implement-entitlement	Implements an entitlement.
do-move-dest-object	Moves an object in the destination data store.
do-move-src-object	Moves an object in the source data store.
do-reformat-op-attr	Changes the format of all values of a particular attribute in the current operation.
do-remove-association	Disassociates an application object.
do-remove-dest-attr-value	Removes a value from an attribute in the destination data store.
do-remove-src-attr-value	Removes a value from an attribute in the source data store.
do-rename-dest-object	Renames an object in the destination data store.
do-rename-op-attr	Changes an attribute name for all operations on that attribute in the current operation.
do-rename-src-object	Renames an object in the source data store.
do-send-email	Generates an e-mail notification.
do-send-email-from-template	Generates an e-mail notification using SMTP configuration and e-mail template objects.
do-set-default-attr-value	Sets the default value for an attribute created in the destination data store
do-set-dest-attr-value	Sets the value of an attribute in the destination data store.
do-set-dest-password	Sets the password for the current object in the destination data store.
do-set-local-variable	Sets the value of a local variable.
do-set-op-association	Sets that association value for the current operation.
do-set-op-class-name	Sets the object class name for the current operation.
do-set-op-dest-dn	Sets the destination DN for the current operation.
do-set-op-property	Sets an operation property.
do-set-op-src-dn	Sets the source DN for the current operation.
do-set-op-template-dn	Sets the template DN for the current add operation.
do-set-src-attr-value	Sets the value of an attribute in the source data store.

Element	Description
do-set-src-password	Sets the password for the current object in the source data store.
do-set-sso-credential	Sets a credential in an SSO credential store.
do-set-sso-passphrase	Sets a passphrase in an SSO credential store.
do-set-xml-attr	Sets custom XML attributes on existing elements.
do-start-workflow	Starts a workflow.
do-status	Reports status.
do-strip-op-attr	Strips an attribute from the current operation.
do-strip-xpath	Strips arbitrary data from the current operation
do-trace-message	Sends a trace message.
do-veto	Vetoes the current operation.
do-veto-if-op-attr-not-available	Vetoes the current operation if a particular attribute is not available in the operation.
do-while	Repeat actions while a condition is true.

Attributes

None

Content Rule

(do-add-association | do-add-dest-attr-value | do-add-dest-object | do-add-src-attr-value | do-add-src-object | do-append-xml-element | do-append-xml-text | do-break | do-clear-dest-attr-value | do-clear-op-property | do-clear-src-attr-value | do-clear-sso-credential | do-clone-op-attr | do-clone-xpath | do-delete-dest-object | do-delete-src-object | do-find-matching-object | do-for-each | do-generate-event | do-if | do-implement-entitlement | do-move-dest-object | do-move-src-object | do-reformat-op-attr | do-remove-association | do-remove-dest-attr-value | do-remove-src-attr-value | do-rename-dest-object | do-rename-op-attr | do-rename-src-object | do-send-email | do-send-email-from-template | do-set-default-attr-value | do-set-dest-attr-value | do-set-dest-password | do-set-local-variable | do-set-op-association | do-set-op-class-name | do-set-op-dest-dn | do-set-op-property | do-set-op-src-dn | do-set-op-template-dn | do-set-src-attr-value | do-set-src-password | do-set-sso-credential | do-set-sso-passphrase | do-set-xml-attr | do-start-workflow | do-status | do-strip-op-attr | do-strip-xpath | do-trace-message | do-veto | do-veto-if-op-attr-not-available | do-while) *

Parent Elements

Element	Description
rule	Rule within a policy.

and

Specifies a set of tests that are performed and whose results are logically ANDed together. A set of <and> elements enclosed by a <conditions> are ORed together.

Example

See <policy>.

Allowed Content

Element	Description
if-association	Tests an association.
if-attr	Tests an attribute in the current operation or the current object in the source data store.
if-class-name	Tests the object class of the current operation.
if-dest-attr	Tests an attribute of the current object in the destination data store.
if-dest-dn	Tests the destination DN of the current operation.
if-entitlement	Tests an entitlement of the current object.
if-global-variable	Tests a global variable.
if-local-variable	Tests a local variable.
if-named-password	Tests a named password.
if-op-attr	Tests an attribute in the current operation.
if-op-property	Tests an operation property.
if-operation	Tests the name of the current operation.
if-password	Tests the password of the current operation.
if-src-attr	Tests an attribute of current object in the source data store.
if-src-dn	Tests the source DN of the current operation.
if-xml-attr	Tests an XML attribute of the current operation.
if-xpath	Tests an XPath expression.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of policy.	false

Content Rule

(if-association | if-attr | if-class-name | if-dest-attr | if-dest-dn | if-entitlement | if-global-variable | if-local-variable | if-named-password | if-op-attr | if-op-property | if-operation | if-password | if-src-attr | if-src-dn | if-xml-attr | if-xpath) *

Parent Elements

Element	Description
arg-conditions	Conditions argument.
conditions	Conditions under which the actions of a <rule> are performed.

arg-actions

Specifies the actions that are performed for each iteration of the enclosing <do-for-each>. It is different from other argument types because it contains actions instead of tokens.

Example

See <do-for-each>, <do-if>, <do-while>, <do-implement-entitlement>.

Allowed Content

Element	Description
do-add-association	Associates the current object.
do-add-dest-attr-value	Adds a value to an attribute in the destination data store.
do-add-dest-object	Adds an object in the destination data store.
do-add-src-attr-value	Adds a value to an attribute in the source data store.
do-add-src-object	Adds an object in the source data store.
do-append-xml-element	Appends a custom XML element to existing elements.
do-append-xml-text	Appends custom XML text to existing elements.
do-break	Stops processing the current operation with this policy.
do-clear-dest-attr-value	Clears all values of an attribute in the destination data store.
do-clear-op-property	Clears an operation property.
do-clear-src-attr-value	Clears all values of an attribute in the source data store.
do-clear-sso-credential	Clears a credential in an SSO credential store.
do-clone-op-attr	Applies all operations on an attribute in the current operation to a different attribute.
do-clone-xpath	Clones and appends a set of nodes to existing elements.
do-delete-dest-object	Deletes an object in the destination data store.
do-delete-src-object	Deletes an object in the source data store.
do-find-matching-object	Automatically associates the current object.
do-for-each	Repeats actions for each node in a node set.
do-generate-event	Generates a user-defined event.
do-if	Conditionally perform actions.
do-implement-entitlement	Implements an entitlement.

Element	Description
do-move-dest-object	Moves an object in the destination data store.
do-move-src-object	Moves an object in the source data store.
do-reformat-op-attr	Changes the format of all values of a particular attribute in the current operation.
do-remove-association	Disassociates an application object.
do-remove-dest-attr-value	Removes a value from an attribute in the destination data store.
do-remove-src-attr-value	Removes a value from an attribute in the source data store.
do-rename-dest-object	Renames an object in the destination data store.
do-rename-op-attr	Changes an attribute name for all operations on that attribute in the current operation.
do-rename-src-object	Renames an object in the source data store.
do-send-email	Generates an e-mail notification.
do-send-email-from-template	Generates an e-mail notification using SMTP configuration and e-mail template objects.
do-set-dest-attr-value	Sets the default value for an attribute created in the destination data store
do-set-dest-attr-value	Sets the value of an attribute in the destination data store.
do-set-dest-password	Sets the password for the current object in the destination data store.
do-set-local-variable	Sets the value of a local variable.
do-set-op-association	Sets that association value for the current operation.
do-set-op-class-name	Sets the object class name for the current operation.
do-set-op-dest-dn	Sets the destination DN for the current operation.
do-set-op-property	Sets an operation property.
do-set-op-src-dn	Sets the source DN for the current operation.
do-set-op-template-dn	Sets the template DN for the current add operation.
do-set-src-attr-value	Sets the value of an attribute in the source data store.
do-set-src-password	Sets the password for the current object in the source data store.
do-set-sso-credential	Sets a credential in an SSO credential store.
do-set-sso-passphrase	Sets a passphrase in an SSO credential store.
do-set-xml-attr	Sets custom XML attributes on existing elements.
do-start-workflow	Starts a workflow.
do-status	Reports status.

Element	Description
do-strip-op-attr	Strips an attribute from the current operation.
do-strip-xpath	Strips arbitrary data from the current operation
do-trace-message	Sends a trace message.
do-veto	Vetoes the current operation.
do-veto-if-op-attr-not-available	Vetoes the current operation if a particular attribute is not available in the operation.
do-while	Repeat actions while a condition is true.

Attributes

None

Content Rule

(do-add-association | do-add-dest-attr-value | do-add-dest-object | do-add-src-attr-value | do-add-src-object | do-append-xml-element | do-append-xml-text | do-break | do-clear-dest-attr-value | do-clear-op-property | do-clear-src-attr-value | do-clear-sso-credential | do-clone-op-attr | do-clone-xpath | do-delete-dest-object | do-delete-src-object | do-find-matching-object | do-for-each | do-generate-event | do-if | do-implement-entitlement | do-move-dest-object | do-move-src-object | do-reformat-op-attr | do-remove-association | do-remove-dest-attr-value | do-remove-src-attr-value | do-rename-dest-object | do-rename-op-attr | do-rename-src-object | do-send-email | do-send-email-from-template | do-set-default-attr-value | do-set-dest-attr-value | do-set-dest-password | do-set-local-variable | do-set-op-association | do-set-op-class-name | do-set-op-dest-dn | do-set-op-property | do-set-op-src-dn | do-set-op-template-dn | do-set-src-attr-value | do-set-src-password | do-set-sso-credential | do-set-sso-passphrase | do-set-xml-attr | do-start-workflow | do-status | do-strip-op-attr | do-strip-xpath | do-trace-message | do-veto | do-veto-if-op-attr-not-available | do-while) *

Parent Elements

Element	Description
do-for-each	Repeats actions for each node in a node set.
do-if	Conditionally perform actions.
do-implement-entitlement	Implements an entitlement.
do-while	Repeats actions while a condition is True.

arg-association

Specifies an association value for the enclosing action. Each of the enclosed tokens is evaluated and the resulting string values are concatenated to form an association value.

Example

See <do-add-association>.

Allowed Content

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.

Element	Description
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

None

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) *

Parent Elements

Element	Description
do-add-association	Associates the current object.
do-add-dest-attr-value	Adds a value to an attribute in the destination data store.
do-add-src-attr-value	Adds a value to an attribute in the source data store.
do-clear-dest-attr-value	Clears all values of an attribute in the destination data store.
do-clear-src-attr-value	Clears all values of an attribute in the source data store.
do-delete-dest-object	Deletes an object in the destination data store.
do-delete-src-object	Deletes an object in the source data store.
do-move-dest-object	Moves an object in the destination data store.
do-move-src-object	Moves an object in the source data store.
do-remove-association	Disassociates an application object.
do-remove-dest-attr-value	Removes a value from an attribute in the destination data store.
do-remove-src-attr-value	Removes a value from an attribute in the source data store.
do-rename-dest-object	Renames an object in the destination data store.
do-rename-src-object	Renames an object in the source data store.
do-set-dest-attr-value	Sets the value of an attribute in the destination data store.
do-set-dest-password	Sets the password for the current object in the destination data store.
do-set-op-association	Sets that association value for the current operation.
do-set-src-attr-value	Sets the value of an attribute in the source data store.
do-set-src-password	Sets the password for the current object in the source data store.

Element	Description
token-dest-attr	The values of an attribute of the current object in the destination data store.
token-query	Queries the source or destination data store.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-src-attr	The values of an attribute of the current object in the source data store.
token-unique-name	Generates a unique name.

arg-component

Provides values for components of the enclosing <arg-value> if the type attribute of <arg-value> is structured. Each of the enclosed tokens is evaluated and the resulting string values are concatenated to form the value of the component. The name of the component is specified by the name attribute.

Example

See <arg-value>.

Allowed Content

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.

Element	Description
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

Attribute	Possible Values	Default Value
name	CDATA Name of the component. Supports variable expansion.	#REQUIRED

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) *

Parent Elements

Element	Description
arg-value	Value argument.

arg-conditions

Specifies the conditions associated with the enclosing action. It is different from other argument types in that it contains conditions instead of tokens.

Example

See <do-if>, <do-while>.

Allowed Content

Element	Description
and	A logical conjunction.
or	A logical disjunction.

Attributes

None

Content Rule

(and * | or *)

Parent Elements

Element	Description
do-if	Conditionally perform actions.
do-while	Repeats actions while a condition is True.

arg-dn

Specifies a DN value for the enclosing action. Each of the enclosed tokens is evaluated and the resulting string values are concatenated to form a DN value.

Example

See <do-add-association>.

Allowed Content

Elements	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or the current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation.
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of the current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current object.
token-dest-name	The unqualified RDN derived from destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.

Elements	Description
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source of destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from the source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	A generated unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

None

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) *

Parent Elements

Element	Description
do-add-association	Associates the current object.
do-add-dest-attr-value	Adds a value to an attribute in the destination data store.
do-add-src-attr-value	Adds a value to an attribute in the source data store.
do-clear-dest-attr-value	Clears all values of an attribute in the destination data store.
do-clear-src-attr-value	Clears all values of an attribute in the source data store.
do-clear-sso-credential	Clears a credential in an SSO credential store.
do-delete-dest-object	Deletes an object in the destination data store.
do-delete-src-object	Deletes an object in the source data store.
do-find-matching-object	Automatically associates the current object.
do-move-dest-object	Moves an object in the destination data store.
do-move-src-object	Moves an object in the source data store.
do-remove-association	Disassociates an application object.
do-remove-dest-attr-value	Removes a value from an attribute in the destination data store.
do-remove-src-attr-value	Removes a value from an attribute in the source data store.
do-rename-dest-object	Renames an object in the destination data store.
do-rename-src-object	Renames an object in the source data store.
do-set-dest-attr-value	Sets the value of an attribute in the destination data store.
do-set-dest-password	Sets the password for the current object in the destination data store.
do-set-op-dest-dn	Sets the destination DN for the current operation.
do-set-op-src-dn	Sets the source DN for the current operation

Element	Description
do-set-op-template-dn	Sets the template DN for the current add operation.
do-set-src-attr-value	Sets the value of an attribute in the source data store.
do-set-src-password	Sets the password for the current object in the source data store.
do-set-sso-credential	Sets a credential in an SSO credential store.
do-set-sso-passphrase	Sets a passphrase in an SSO credential store.
do-start-workflow	Starts a workflow.
token-dest-attr	The values of an attribute of the current object in the destination data store.
token-query	Queries the source or destination data store.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-src-attr	The values of an attribute of the current object in the source data store.
token-unique-name	Generates a unique name.

arg-match-attr

Specifies the attributes to be used to find a match for the enclosing <do-find-matching-object>. The name attribute provides the name of the attribute to use for matching. If there is an enclosed <arg-value>, then it provides the attribute value to use for matching, otherwise the values are from the values available in the current operation.

Example

See <do-find-matching-object>.

Allowed Content

Element	Description
arg-value	An argument value.

Attributes

Attribute	Possible Values	Default Value
name	CDATA Name of the attribute.	#REQUIRED

Content Rule

(arg-value ?)

Parent Elements

Element	Description
do-find-matching-object	Automatically associates the current object.
token-query	Queries the source or destination data store.

arg-node-set

Specifies an XPath 1.0 node set for the enclosing action. Each of the enclosed tokens are evaluated and if the token returns a node set then the nodes in that set are added to the result set otherwise, a text node is created and added to the node set.

Example

See <do-for-each>.

Allowed Content

Elements	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or the current object in the source data source.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation.
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of the current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.

Elements	Description
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses and converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an entitlement revoked in the current operation.
token-removed-entitlement	The values of an attribute removed in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from the source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	A generated unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

None

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath | token-query | token-split) *

Parent Elements

Element	Description
do-for-each	Repeats actions for each node in a node set.
do-implement-entitlement	Implements an entitlement.
do-set-local-variable	Sets the value of a local variable.

arg-object

Specifies a Java object for storing in the local variable specified by the enclosing <do-set-local-variable> action. The enclosed token must a <token-xpath> that specifies an expression that returns a Java object or a <token-local-variable> for a variable that already contains a Java object.

Example

See <do-set-local-variable>.

Allowed Content

Element	Description
token-local-variable	The value of a local variable.
token-xpath	The result of an XPath expression.

Attributes

None

Content Rule

(token-local-variable | token-xpath)

Parent Elements

Element	Description
do-set-local-variable	Sets the value of a local variable.

arg-password

Specifies a password to be used by the enclosing action. Each of the enclosed tokens is evaluated and the resulting string values are concatenated to form a string value.

Example

See <do-start-workflow>.

Allowed Content

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or the current object in the source data source.
token-base64-decode	Decodes base64 data into string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation.
token-convert-time	Converts date/time from one format to another.
token-dest-attr	The values of an attribute of the current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lower case.
token-map	Maps a string through a mapping table.

Element	Description
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from the source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	A generated unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to upper case.
token-xml-parse	Parses XML.
token-xml-serialize	Serialize XML.
token-xpath	The result of an XPath expression.

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.

Element	Description
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.

Element	Description
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

None

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) *

Parent Elements

Element	Description
do-send-email	Generates an e-mail notification.
do-send-email-from-template	Generates an e-mail notification using SMTP configuration and e-mail template objects.
do-start-workflow	Starts a workflow.

arg-string

Specifies string value for the enclosing action. Each of the enclosed tokens is evaluated and the resulting string values are concatenated to form a string value.

Example

See <do-set-op-class-name>.

Allowed Content

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.

Element	Description
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

Attribute	Possible Values	Default Value
name	CDATA Name of the argument. Supports variable expansion.	#IMPLIED

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) *

Parent Elements

Element	Description
do-append-xml-text	Appends custom XML text to existing elements.
do-clear-sso-credential	Clears a credential in an SSO credential store.
do-generate-event	Generates a user-defined event.
do-rename-dest-object	Renames an object in the destination data store.
do-rename-src-object	Renames an object in the source data store.
do-send-email	Generates an e-mail notification.
do-send-email-from-template	Generates an e-mail notification using SMTP configuration and e-mail template objects.
do-set-dest-password	Sets the password for the current object in the destination data store.
do-set-local-variable	Sets the value of a local variable.
do-set-op-class-name	Sets the object class name for the current operation.
do-set-op-property	Sets an operation property.
do-set-src-password	Sets the password for the current object in the source data store.
do-set-sso-credential	Sets a credential in an SSO credential store.
do-set-sso-passphrase	Sets a passphrase in an SSO credential store.
do-set-xml-attr	Sets custom XML attributes on existing elements.

Element	Description
do-start-workflow	Starts a workflow.
do-status	Reports status.
do-trace-message	Sends a trace message.
token-document	Reads an XML document.
token-query	Queries the source or destination data store.
token-unique-name	Generates a unique name.

arg-value

Specifies an attribute value for the enclosing action. If the type attribute is structured, then the content of <arg-value> must be a set of <arg-component> elements. If the type attribute is other than structured, then each of the enclosed tokens is evaluated and the resulting string values are concatenated to form a value.

Example

```
<arg-value>
  <token-attr name="Surname"/>
  <token-text>, </token-text>
  <token-attr name="Given Name"/>
</arg-value>
<arg-value type="structured">
  <arg-component name="string">
  <token-text>EN</token-text>
  </arg-component>
    <arg-component name="string">
    <token-text>JP</token-text>
  </arg-component>
</arg-value>
```

Allowed Content

Element	Description
arg-component	Component argument.
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.

Element	Description
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.

Element	Description
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

Attribute	Possible Values	Default Value
type	CDATA Type of the value. Supports variable expansion.	string

Content Rule

(arg-component + | (token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) *

Parent Elements

Element	Description
arg-match-attr	Matches an attribute argument.
do-add-dest-attr-value	Adds a value to an attribute in the destination data store.
do-add-src-attr-value	Adds a value to an attribute in the source data store.
do-reformat-op-attr	Changes the format of all values of a particular attribute in the current operation.
do-remove-dest-attr-value	Removes a value from an attribute in the destination data store.
do-remove-src-attr-value	Removes a value from an attribute in the source data store.
do-set-default-attr-value	Sets the default value for an attribute to be created in the destination data store.
do-set-dest-attr-value	Sets the value of an attribute in the destination data store.
do-set-src-attr-value	Sets the value of an attribute in the source data store.

comment

A long description or other textual information relating to the containing <rule>. It does not affect the execution of the <rule>.

Remarks

A comment has a name that can have special meaning to a user interface agent that displays or edits the rule. Policy Builder currently supports one instance per rule of an unnamed comment, and one instance each of comments with the names author, version, and lastChanged. Additional named and unnamed comments are allowed but are ignored by Policy Builder.

Example

See <policy>.

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
name	CDATA Name of the comment.	#IMPLIED

Content Rule

(#PCDATA)

Parent Elements

Element	Description
rule	Rule within a policy.

component

Provides values for components of the enclosing if-condition if the mode attribute of that conditions is structured.

Example

See <if-attr>.

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
name	CDATA Name of the component. Supports variable expansion.	#REQUIRED

Content Rule

(#PCDATA)

Parent Elements

Element	Description
if-attr	Tests an attribute in the current operation or the current object in the source data store.
if-dest-attr	Tests an attribute of the current object in the destination data store.
if-op-attr	Tests an attribute in the current operation.
if-src-attr	Tests an attribute of the current object in the source data store.

conditions

The conditions under which the <actions> of the enclosing <rule> are performed. The <conditions> are always specified in Conjunctive Normal Form (CNF) or Disjunctive Normal Form (DNF). As such, the content of <conditions> is either a disjunction of conjunctions specified by a (possibly empty) set of <and> elements or a conjunction of disjunctions specified by a (possibly empty) set of <or> elements. The <actions> of the enclosing <rule> are only performed when the logical expression represented in CNF or DNF evaluates to true or when no conditions are specified.

Remarks

The evaluation of the conditions uses short-circuit logic so that no additional tests are performed when it is possible to determine the resultant Boolean value of the <conditions>.

All individual condition tests are represented by an element of the form <if-* op="some operator">.

Some condition tests have a mode parameter that indicates the algorithm to use for comparisons. The following table details the modes that are available.

Element	Description
case	Character-by-character case-sensitive comparison.
nocase	Character-by-character case-insensitive comparison.
regex	Regular expression match of the entire string. Case-insensitive by default, but can be changed by an escape in the expression. See http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Pattern.html and http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Matcher.html#matches(). Pattern options CASE_INSENSITIVE, DOTALL, and UNICODE_CASE are used but can be reversed using the appropriate embedded escapes.
src-dn	Compares using semantics appropriate to the DN format for the source data store.
dest-dn	Compares using semantics appropriate to the DN format for the destination data store.
numeric	Compares numerically.
octet	Compares octet (Base64-encoded) values.
structured	Compares structured attributes according to the comparison rules for the structured syntax of the attribute.

Example

See <policy>.

Allowed Content

Element	Description
and	Logical conjunction.
or	Logical disjunction.

Attributes

None

Content Rule

(and * | or *)

Parent Elements

Element	Description
rule	Rule within a policy.

description

A description of the containing <rule> or <policy>. It does not affect the execution of the <rule> or <policy>.

Example

See <policy>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
policy	A policy.
rule	Rule within a policy.

do-add-association

Sends an <add-association> command to eDirectory. The dest-dn for the command comes from the value <arg-dn> if it is specified, or from the current object if not specified. The association value sent is provided by <arg-association>.

Example

```
<do-add-association>
  <arg-dn>
    <token-src-dn/>
  </arg-dn>
  <arg-association>
    <token-src-name/>
  </arg-association>
</do-add-association>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-association	Association argument.

Attributes

Attribute	Possible Values	Default Value
direct	true | false Use destCommandProcessor to carry out this action. Deprecated. Use when="direct" instead.	false
disabled	true | false True if this element is disabled.	false

Attribute	Possible Values	Default Value
when	auto | before | after | direct When this action should be performed: auto - Automatically determined (either in or after the current operation). before - Before the current operation. after - After the current operation. direct - Written directly to the destination data store instead of being added to the current document.	auto

Content Rule

(arg-dn ? , arg-association)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-add-dest-attr-value

Adds the value specified by <arg-value> to the named attribute on an object in the destination data store. The target object is specified by either <arg-dn> or <arg-association> if specified, or by the current object if not specified.

Example

```
<do-add-dest-attr-value name="Member">
  <arg-dn>
    <token-text>Users/ManagerGroup</token-text>
  </arg-dn>
  <arg-value>
    <token-dest-dn/>
  </arg-value>
</do-add-dest-attr-value>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-association	Association argument.
arg-value	Value argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA Class name of object to modify. Can be required (for schema mapping purposes) if the object is other than the current object. Supports variable expansion.	#IMPLIED
direct	true | false Use destCommandProcessor to carry out this action. Deprecated. Use when="direct" instead.	false
disabled	true | false True if this element is disabled.	false
name	CDATA Name of the attribute. Supports variable expansion.	#REQUIRED

Attribute	Possible Values	Default Value
when	auto | before | after | direct When this action should be performed: auto - Automatically determined (either in or after the current operation). before - Before the current operation. after - After the current operation. direct - Written directly to the destination data store instead of being added to the current document.	auto

Content Rule

((arg-dn | arg-association) ? , arg-value)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Association argument.

do-add-dest-object

Creates an object of type class-name that in the destination data store with a name and location provided by <arg-dn>. Any attribute values added as part of the object creation must be done in subsequent <do-add-dest-attr-value> actions using the same <arg-dn>.

Example

```
<do-add-dest-object class-name="User">
  <arg-dn>
    <token-text>Users/Fred Flintstone</token-text>
  </arg-dn>
</do-add-dest-object>
<do-add-dest-attr-value name="Surname">
  <arg-dn>
    <token-text>Users/Fred Flintstone</token-text>
  </arg-dn>
  <arg-value>
    <token-text>Flintstone</token-text>
  </arg-value>
</do-add-dest-attr-value>
```

Allowed Content

Element	Description
arg-dn	DN argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA	#REQUIRED
	Class name of the object to create. Supports variable expansion.	
direct	true | false	false
	Use destCommandProcessor to carry out this action.	
	Deprecated. Use when="direct" instead.	
disabled	true | false	false
	True if this element is disabled.	

Attribute	Possible Values	Default Value
when	auto | before | after | direct When this action should be performed: auto - Automatically determined (either in or after the current operation). before - before the current operation. after - After the current operation. direct - Written directly to the destination data store instead of being added to the current document.	auto

Content Rule

(arg-dn)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-add-role

Initiates a request to the Roles Based Provisioning Module (RBPM) to assign the Role specified by role-id to an Identity.

Remarks

The target Identity is specified by either <arg-dn> or <arg-association> if specified or by the current object otherwise. If specified by <arg-dn>, the DN must be in the LDAP format. If the target identity is specified by either <arg-dn> or <arg-association>, then the role-assignment-type must be specifed from one of USER_TO_ROLE, GROUP_TO_ROLE, CONTAINER_TO_ROLE or ROLE_TO_ROLE. If the role-assignment-type is not specified, then the assignment type is defaulted to USER_TO_ROLE. The request is made to the RBPM enabled User Application server specified by url using credentials specified by id and <arg-password>. Additional optional arguments to the Role assignment request might be specified by the following named <arg-string>s.

Name	Description
role-assignment-type	The role assignment type from one of from one of USER_TO_ROLE, GROUP_TO_ROLE, CONTAINER_TO_ROLE or ROLE_TO_ROLE. Default: USER_TO_ROLE
description	A description of the reason for the request used for auditing and (if necessary) approval purposes. Default: Request generated by policy.
effective-time	The time (in CTIME format) the role assignment should become effective. Default: now
expiration-time	The time (in CTIME format) the role assignment automatically expires. Default: never
sod-justification	A justification for requesting an exception for any Separation of Duty violations this assignment will trigger. Default: No exception will be requested and the request will fail if it causes a violation.

If any type of error occurs while requesting the role assignment, the error string is available to the enclosing policy in the local variable named *error.do-add-rol*e. Otherwise that local variable is not available.

Example

```
<do-add-role
    id="cn=RoleAdmin,o=People"
  url="http://localhost:8080/IDMProv"
role-
id="cn=Contractor,cn=Level30,cn=RoleDefs,cn=RoleConfig,cn=AppConfig,cn=UserApplica
tion,cn=DriverSet,o=novell"
  time-out="5000">
  <arg-password>
    <token-named-password name="role-admin"/>
  </arg-password>
  <arg-dn>
    <token-text xml:space="preserve">cn=Contractors,ou=Groups,o=Data</token-text>
  </arg-dn>
  <arg-string name="role-assignment-type">
    <token-text>GROUP_TO_ROLE</token-text>
  </arg-string>
  <arg-string name="description">
    <token-text>Requested by policy because isContractor attribute set to true</
token-text>
  </arg-string>
  <arg-string name="effective-time">
    <token-src-attr name="Hire Date"/>
  </arg-string>
  <arg-string name="expiration-time">
    <token-convert-time dest-format="!CTIME" dest-tz="UTC" offset="6" offset-
unit="month" src-format="!CTIME" src-tz="UTC">
    <token-src-attr name="Hire Date"/>
</token-convert-time>
  </arg-string>
</do-add-role
```

Allowed Content

Element	Description
arg-password	Password argument.
arg-dn	DN argument.
arg-association	Association argument.
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
id	CDATA The LDAP format DN of a user authorized to make the request. Supports variable expansion.	#REQUIRED

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false
role-id	CDATA The LDAP format DN of the Role to assign. Supports variable expansion.	#REQUIRED
time-out	CDATA The number of milliseconds you want Identity Manager to try to establish a connection to the User Application server before timing out. Supports variable expansion.	0
url	CDATA The URL of the User Application server hosting the Roles Based Provisioning Module. Supports variable expansion.	#REQUIRED

Content Rule

(arg-password, (arg-dn | arg-association) ? , arg-string *)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-add-resource

Initiates a request to the Roles Based Provisioning Module (RBPM) to assign the Resource specified by resource-id to an Identity.

Remarks

The target Identity is specified by either <arg-dn> or <arg-association> if specified or by the current object otherwise. If specified by <arg-dn>, the DN must be in the LDAP format. The request is made to the RBPM enabled User Application server specified by url using credentials specified by id and <arg-password>. Additional optional arguments to the Resource assignment request might be specified by the following named <arg-string>s.

Name	Description
description	A description of the reason for the request used for auditing and (if necessary) approval purposes. Default: Request generated by policy.

NOTE: You can specify parameter values for the added resources. You can use the plus sign (+) to insert a new string, or select the *Edit the Strings* icon to open the String Builder and specify the strings. You must specify the parameter names as param1, param 2 and so on. For more information about the Named String Builder, see "String Builder" in *Policies in Designer 4.0.2*.

If any type of error occurs while requesting the resource assignment, the error string is available to the enclosing policy in the local variable named *error.do-add-resource*. Otherwise that local variable is not available.

Example

```
<do-add-resource
    id="CN=UAAdmin,OU=Sa,O=Data"
  url="http://localhost:8080/IDMProv"
  resource-
id="CN=Computer,CN=ResourceDefs,CN=RoleConfig,CN=AppConfig,CN=UserApplication,CN=D
riverSet,O=System"
  time-out="5000">
  <arg-password>
    <token-named-password name="resource-admin"/>
  </arg-password>
  <arg-string name="description">
    <token-text>Requested by policy because requireComputer attribute set to true</
token-text>
  </arg-string>
</do-add-resource>
```

Allowed Content

Element	Description
arg-password	Password argument.
arg-dn	DN argument.
arg-association	Association argument.
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
id	CDATA The LDAP format DN of a user authorized to make the request. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false
requester-id	CDATA The LDAP format DN of a user who makes the request. Supports variable expansion.	#REQUIRED
resource-id	CDATA The LDAP format DN of the Resource to assign. Supports variable expansion.	#REQUIRED
time-out	CDATA The number of milliseconds you want Identity Manager to try to establish a connection to the User Application server before timing out. Supports variable expansion.	0
url	CDATA The URL of the User Application server hosting the Roles Based Provisioning Module. Supports variable expansion.	#REQUIRED

Content Rule

(arg-password, (arg-dn | arg-association) ? , arg-string *)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Association argument.

do-add-src-attr-value

Adds the value specified by <arg-value> to the named attribute on an object in the source data store. The target object is specified by either <arg-dn> or <arg-association> if specified, or by the current object if not specified.

Example

```
<do-add-src-attr-value name="Member">
  <arg-dn>
    <token-text>Users/ManagerGroup</token-text>
  </arg-dn>
  <arg-value>
    <token-dest-dn/>
  </arg-value>
</do-add-src-attr-value>
```

Allowed Content

Element	Description
arg-password	DN argument.
arg-association	Association argument.
arg-value	Value argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA Class name of the object to modify. an be required (for schema mapping purposes) if object is other than the current object. Supports variable expansion.	#IMPLIED
disabled	true | false True if this element is disabled.	false
name	CDATA Name of the attribute. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

((arg-dn | arg-association) ? , arg-value)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Association argument.

do-add-src-object

Creates an object of type class-name in the source data store with a name and location provided by <arg-dn>. Any attribute values to be added as part of the object creation must be done in subsequent <do-add-src-attr-value> actions using the same <arg-dn>.

Example

```
<do-add-src-object class-name="User">
  <arg-dn>
    <token-text>Users/Fred Flintstone</token-text>
  </arg-dn>
</do-add-src-object>
<do-add-src-attr-value name="Surname">
  <arg-dn>
    <token-text>Users/Fred Flintstone</token-text>
  </arg-dn>
  <arg-value>
    <token-text>Flintstone</token-text>
  </arg-value>
</do-add-src-attr-value>
```

Allowed Content

Element	Description
arg-dn	DN argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA	#REQUIRED
	Class name of object to create	
disabled	true | false	false
	True if this element is disabled.	
notrace	true | false	false
	True if this element should not be traced during execution of policy.	

Content Rule

(arg-dn)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-append-xml-element

Appends a custom element named by the name attribute to the set of elements selected by an expression.

Example

```
<do-append-xml-element name="jdbc:statement" expression=".."/>
<do-append-xml-element name="jdbc:sql"
     expression="../jdbc:statement[last()]"/>
<do-append-xml-text expression="../jdbc:statement[last()]/jdbc:sql">
   <arg-string>
      <token-text> UPDATE dirxml.emp SET fname = '</token-text>
      <token-op-attr name="Given Name"/>
      <token-text>' </token-text>
   </arg-string>
</do-append-xml-text>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
before	CDATA XPath 1.0 expression evaluated relative to each of the nodes select by expression that returns a node-set containing the child nodes which the new elements should be inserted before.	#IMPLIED
disabled	true | false True if this element is disabled.	false
expression	CDATA XPath 1.0 expression that returns a node-set containing the elements to which the new elements should be appended.	#REQUIRED
name	NMTOKEN Tag name of the element Can contain a namespace prefix if that prefix has been defined on the <policy>.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-append-xml-text

Appends the text provided by <arg-string> to the set of elements selected by expression.

Example

See <do-append-xml-element>.

Allowed Content

Element	Description
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
before	CDATA XPath 1.0 expression evaluated relative to each of the nodes select by expression that returns a node-set containing the child nodes which the text should be inserted before.	#REQUIRED
disabled	true | false True if this element is disabled.	false
expression	CDATA XPath 1.0 expression that returns a node-set containing the elements to which the new elements should be appended.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(arg-string)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-break

Stops the current operation from being processed by any more actions or rules within the current policy.

Example

```
<do-break/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-clear-dest-attr-value

Removes all the values for the named attribute from an object in the destination data store. The target object is specified by either <arg-dn> or <arg-association> if specified, or by the current object if not specified.

Example

```
<do-clear-dest-attr-value name="Member">
   <arg-dn>
      <token-text>Users/ManagerGroup</token-text>
   </arg-dn>
</do-clear-dest-attr-value>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-association	Association argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA Class name of object to modify. Can be required (for schema mapping purposes) if object is other than the current object. Supports variable expansion.	#IMPLIED
direct	true | false Use destCommandProcessor to carry out this action. Deprecated. Use when="direct" instead.	false
disabled	true | false True if this element is disabled.	false
name	CDATA Name of the attribute. Supports variable expansion.	#REQUIRED

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false
when	auto | before | after | direct When this action should be performed: auto - Automatically determined (either in or after the current operation). before - Before the current operation. after - After the current operation. direct - Written directly to the destination data store instead of being added to the current document.	auto

Content Rule

(arg-dn | arg-association) ?

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-clear-op-property

Clears any operation property with the given name from the current operation.

Example

```
<do-clear-op-property name="myProperty"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
name	NMTOKEN Name of the operation property.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-clear-src-attr-value

Removes all values for the named attribute from an object in the source data store. The target object is specified by either <arg-dn> or <arg-association> if specified, or by the current object if not specified.

Example

```
<do-clear-src-attr-value name="Member">
   <arg-dn>
      <token-text>Users/ManagerGroup</token-text>
   </arg-dn>
</do-clear-src-attr-value>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-association	Association argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA	#IMPLIED
	Class name of the object to modify.	
	Can be required (for schema mapping purposes) if object is other than the current object.	
disabled	true | false	false
	True if this element is disabled.	
name	CDATA	#REQUIRED
	Name of the attribute.	
notrace	true | false	false
	True if this element should not be traced during execution of the policy.	

Content Rule

(arg-dn | arg-association) ?

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-clear-sso-credential

Clears a credential from the object specified by the <arg-dn> element in the Single Sign On credential store specified by the <store-def-dn> element for the application specified by the <app-id> element. Additional information about the credential to be cleared can be specified by additional named <arg-string> elements. The number of the strings and the names used are dependent on the credential store and application for which the credential is targeted.

Example

```
<do-clear-sso-credential store-def-dn="../Library/SSO1" app-id="AD7">
 <arg-dn>
  <token-parse-dn src-dn-format="src-dn" dest-dn-format="ldap" start="0" length="-
1">
   <token-src-dn/>
  </token-parse-dn>
 </arg-dn>
</do-clear-sso-credential>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
app-def-dn	CDATA DN of the application credential definition object. Only used by the UI so the various UIs should agree on the DN format used.	#IMPLIED
app-id	CDATA Application ID for the credential. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Attribute	Possible Values	Default Value
store-def-dn	CDATA Slash form DN of the credential store definition object. Can be relative to the including policy. Supports variable expansion.	#REQUIRED

Content Rule

(arg-dn , arg-string *)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule> element.
arg-actions	Actions argument.

do-clone-op-attr

Duplicates all elements that are children of the current operation with the attr-name attribute equal to the name specified by src-name within the operation with attr-name set to dest-name.

Example

```
<do-clone-op-attr src-name="Member" dest-name="Equivalent to Me"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
dest-name	CDATA The attribute name to give to the clone.	#REQUIRED
disabled	true | false True if this element is disabled.	false
notrace t	rue | false True if this element should not be traced during execution of the policy.	false
src-name	CDATA The attribute name to clone.	#REQUIRED

Content Rule

Empty

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-clone-xpath

Appends deep copies of the nodes selected by <src-expression> to the set of elements selected by <dest-expression>.

Example

```
<do-append-xml-element name="delete" expression=".."/>
<do-clone-xpath src-expression="@*" dest-expression="../modify[last()]"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
before	CDATA	#REQUIRED
	XPath 1.0 expression evaluated relative to each of the nodes select by dest-expression that returns a node-set containing the child nodes which the non-attribute cloned nodes should be inserted before.	
dest-expression .	CDATA	#REQUIRED
	XPath 1.0 expression that returns a node-set containing the elements to which the cloned nodes should be appended	
disabled	true | false	false
	True if this element is disabled.	
notrace	true | false	false
	True if this element should not be traced during execution of the policy.	
src-expression	CDATA	#REQUIRED
	XPath 1.0 expression that returns a node-set containing the nodes that are cloned	

Content Declaration

Empty

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-delete-dest-object

Deletes an object in the destination data store. The target object is specified by either <arg-dn> or <arg-association> if specified or by the current object otherwise.

Example

```
<do-delete-dest-object>
   <arg-dn>
      <token-text>Users/Fred Flintstone</token-text>
   </arg-dn>
</do-delete-dest-object>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-association	Association argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA Class name of target object. Support variable expansion.	#IMPLIED
direct	true | false Use destCommandProcessor to carry out this action. Deprecated. Use when="direct" instead.	false
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Attribute	Possible Values	Default Value
when	auto | before | after | direct When this action should be performed: auto - Automatically determined (either in or after the current operation). before - Before the current operation. after - After the current operation. direct - Written directly to the destination data store instead of being added to the current document.	auto

Content Rule

((arg-dn | arg-association) ?)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-delete-src-object

The <do-delete-src-object> action deletes the object in the source data store. The target object is specified by either <arg-dn> or <arg-association> if specified, or by the current object if not specified.

Example

```
<do-delete-src-object>
   <arg-dn>
      <token-text>Users/Fred Flintstone</token-text>
   </arg-dn>
</do-delete-src-object>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-association	Association argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA Class name of target object. Support variable expansion.	#IMLPIED
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

((arg-dn | arg-association) ?)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-find-matching-object

Performs a query in the destination data store and in the dest-dn or the dest-dn <association> to be added to the current operation. It is only valid when the current operation is <add>.

Remarks

<arg-dn> is required when scope="entry" and optional otherwise.

At least one <arg-match-attr> is required when scope="subtree" or scope="subordinates". Because it is undefined what <query> does with <search-attr> when scope="entry", it is also undefined what <do-find-matching-object> does.

The <query> generates a scope attribute based on the scope attribute of the <do-find-matching-object>. It is a dest-dn attribute set to the content of <arg-dn>, if any. It is a class-name attribute and <search-class> based on the class-name attribute from the current object. For each <arg-match-attr> there is a <search-attr> for the same attribute, populated with either the <arg-value> content of <arg-match-attr> (if it exists) or the values available in the current operation. If no value is available, then no query is performed and the action does not find a match.

Any <instance> elements returned from the <query> are considered matches.

If the destination data store is the application, then an association is added to the current operation for each <instance> that is returned. No query is performed if the current operation already has a non-empty association, thus allowing multiple <do-find-matching-object> actions to be strung together in the same rule. If more than one <instance> is returned, then the local variable *error.do-find-matching-object* will be set to a node-set containing the list of src-dn's from the instances if they are available, or the list of associations if the src-dn's are not available.

If the destination data store is eDirectory, then the dest-dn attribute for the current operation is set. No query is performed if the current operation already has a non-empty dest-dn attribute, thus allowing multiple <do-find-matching-object> actions to be strung together in the same rule. If only a single <instance> is returned and that <instance> is not already associated, then the dest-dn of the current operation is set to the src-dn of the <instance> and the local variable *error.do-find-matching-object* is not available. If only a single <instance> is returned and that <instance> is already associated, then the dest-dn of the current operation is set to the single character ￼ and the local variable *error.do-find-matching-object* is set to the src-dn from that <instance>. If multiple <instance> elements are returned then the dest-dn of the current operation is set to the single character &#xFFFD; and the local variable *error.do-find-matching-object* is set to a node-set containing the src-dn's from those <instance>'s..

Example

```
<do-find-matching-object scope="subordinates">
   <arg-dn>
     <token-text>Users/</token-text>
     <token-attr name="OU"/>
   </arg-dn>
   <arg-match-attr name="CN"/>
   <arg-match-attr name="L"/>
   <arg-value>
      <token-text>Provo</token-text>
   </arg-value>
</do-find-matching-object>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-match-attr	Match attribute argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false
scope	entry | subordinates | subtree The scope to be searched.	subtree

Content Rule

((arg-dn ? , arg-match-attr +) | (arg-dn , arg-match-attr *))

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-for-each

Repeats the actions specified by <arg-actions> once for each node in <arg-node-set> with the local variable current-node set to a node-set containing only that node. If the current-node is <entitlement-impl>, then the actions are also marked as if they were also enclosed in <do-implement-entitlement>.

Example

```
<do-for-each>
   <arg-node-set>
      <token-added-entitlement name="Group">
   </arg-node-set>
   <arg-actions>
      <do-add-dest-attr-value name="Member" class-name="Group">
         <arg-dn>
            <token-local-variable name="current-node"/>
         </arg-dn>
         <arg-value type="dn">
            <token-dest-dn/>
         </arg-value>
     </do-add-dest-attr-value>
   </arg-actions>
</do-for-each>
```

Allowed Content

Element	Description
arg-node-set	Node set argument.
arg-actions	Actions argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(arg-node-set , arg-actions)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-generate-event

Sends an Identity Manager user-defined event to the Novell Auditing service. Identity Manager user defined event IDs must be between the range of 1000 to 1999. Valid event levels are defined in the table below. The remaining event data fields are provided by four <arg-string> elements with name attributes. The Novell Audit event structure contains two strings (text1, text2) along with one integer (value) and generic field (data). The two text fields are limited to 256 bytes and the data field can contain up to 3 KB of information.

Remarks

Level	Description
log-emergency	Events that cause the Metadirectory engine or Identity Manager driver to shut down.
log-alert	Events that require immediate attention.
log-critical	Events that can cause parts of the Metadirectory engine or Identity Manager driver to malfunction.
log-error	Events describing errors that can be handled by the Metadirectory engine or Identity Manager driver.
log-warning	Negative events not representing a problem.
log-notice	Events (positive or negative) an administrator can use to understand or improve use and operation.
log-info	Positive events of any importance.
log-debug	Events of relevance for support or engineers to debug operation of the Metadirectory engine or Identity Manager driver.

Tag	Description
text1	Text entered here is stored in the text1 event field.
text2	Text entered here is stored in the text2 event field.
value	Any number entered here is stored in the value1 event field.
data	Data entered here is stored in the blob event field.

Example

```
<do-generate-event id="1000" level="log-info">
   <arg-string name="text1">
      <token-text>User defined data for text1 field</token-text>
   </arg-string>
   <arg-string name="text2">
      <token-text>User defined data for text2 field</token-text>
   </arg-string>
   <arg-string name="value">
      <token-text>-602</token-text>
   </arg-string>
   <arg-string name="data">
      <token-text>User defined blob data</token-text>
   </arg-string>
</do-generate-event >
```

Allowed Content

Element	Description
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
id	CDATA Identity Manager user-defined event ID (1000-2000). Supports variable expansion.	#REQUIRED
level	log-emergency | log-alert | log-critical | log-error | log-warning | log-notice | log-info | log-debug Novell Audit log level.	log-info
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(arg-string *)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-if

Causes the actions specified by the first <arg-actions> to be performed if the conditions specified by <arg-conditions> evaluate to true or the actions specified by the second <arg-actions> (if it exists) to be performed if the conditions specified by <arg-conditions> evaluate to false.

Example

```
<do-if>
 <arg-conditions>
  <and>
   <if-op-attr name="Given Name" op="equal" mode="nocase">fred</if-op-attr>
  </and>
 </arg-conditions>
 <arg-actions>
  <do-set-dest-attr-value name="Surname">
   <arg-value type="string">
    <token-text>Flintstone</token-text>
   </arg-value>
  </do-add-dest-attr-value>
 </arg-actions>
 <arg-actions>
  <do-set-dest-attr-value name="Surname">
   <arg-value type="string">
    <token-text>Rubble</token-text>
   </arg-value>
  </do-add-dest-attr-value>
 </arg-actions>
</do-if>
```

Allowed Content

Element	Description
arg-conditions	Conditions argument.
arg-actions	Actions argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(arg-conditions , arg-actions , arg-actions ?)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-implement-entitlement

Marks the actions specified by <arg-actions> with the <entitlement-impl> elements specified in <arg-node-set> so that the Metadirectory engine knows to report the results of those actions to the DirXML-EntitlementResult attribute of the current object.

Example

```
<do-implement-entitlement>
   <arg-node-set>
      <token-removed-entitlement name="Account"/>
   </arg-node-set>
   <arg-actions>
      <do-set-dest-attr-value name="Login Disabled">
         <arg-value type="state">
            <token-text >true</token-text>
         </arg-value>
      </do-set-dest-attr-value>
   </arg-actions>
</do-implement-entitlement>
```

Allowed Content

Element	Description
arg-node-set	Node set argument.
arg-actions	Actions argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false	false
	True if this element is disabled.	
notrace	true | false	false
	True if this element should not be traced during execution of the policy.	

Content Rule

(arg-node-set , arg-actions)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-move-dest-object

Moves an object in the destination data store. If two arguments are provided, the object identified by the first argument is moved to the container identified by the second argument. If only a single argument is provided, the current object is moved to the container identified by the single argument.

Example

```
<do-move-dest-object>
   <arg-dn>
      <token-text>Users/Active/FredFlintstone</token-text>
   </arg-dn>
   <arg-dn>
      <token-text>Users/InActive</token-text>
   </arg-dn>
</do-move-dest-object>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-association	Association argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA Class name of target object. Supports variable expansion.	#IMPLIED
direct	true | false Use destCommandProcessor to carry out this action. Deprecated. Use when="direct" instead.	false
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Attribute	Possible Values	Default Value
when	auto | before | after | direct When this action should be performed: auto - Automatically determined (either in or after the current operation). before - Before the current operation. after - After the current operation. direct - Written directly to the destination data store instead of being added to the current document.	auto

Content Rule

((arg-dn | arg-association) ? , (arg-dn | arg-association))

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-move-src-object

Moves an object in the source data store. If two arguments are provided then the object identified by the first argument is moved to the container identified by the second argument. If only a single argument is provided, then the current object is moved to the container identified by the single argument.

Example

```
<do-move-src-object>
   <arg-dn>
      <token-text>Users/Active/FredFlintstone</token-text>
   </arg-dn>
   <arg-dn>
      <token-text>Users/InActive</token-text>
   </arg-dn>
</do-move-src-object>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-association	Association argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA Class name of target object. Supports variable expansion.	#IMPLIED
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

((arg-dn | arg-association) ? , (arg-dn | arg-association))

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-reformat-op-attr

Replaces all values for the named attribute within the current operation with the <arg-value>. The value of <arg-value> is evaluated once for each value being replaced with the local variable current-value set to the original value.

Example

```
<do-reformat-op-attr name="CN">
   <arg-value>
      <token-upper-case>
         <token-local-variable name="current-value"/>
      </token-upper-case>
   </arg-value>
</do-reformat-op-attr>
<do-reformat-op-attr name="EMail Address">
   <arg-value>
      <token-xpath expression="$current-value/component[@name='eMailAddr']"/>
   </arg-value>
</do-reformat-op-attr>
```

Allowed Content

Element	Description
arg-value	Value argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
name	CDATA Name of the attribute.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(arg-value)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-remove-association

Sends a <remove-association> command to eDirectory. The association value sent is provided by <arg-association>.

Example

```
<do-remove-association>
   <arg-association>
      <token-src-name/>
   </arg-association>
</do-remove-association>
```

Allowed Content

Element	Description
arg-association	Association argument.

Attributes

Attribute	Possible Values	Default Value
direct	true | false Use destCommandProcessor to carry out this action. Deprecated. Use when="direct" instead.	false
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Attribute	Possible Values	Default Value
when	auto | before | after | direct When this action should be performed: auto - Automatically determined (either in or after the current operation). before - Before the current operation. after - After the current operation. direct - Written directly to the destination data store instead of being added to the current document.	auto

Content Rule

(arg-association)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-remove-dest-attr-value

Removes the value specified by <arg-value> from the named attribute on an object in the destination data store. The target object is specified by either <arg-dn> or <arg-association> if specified or by the current object otherwise.

Example

```
<do-remove-dest-attr-value name="Member">
   <arg-dn>
      <token-text>Users/ManagerGroup</token-text>
   </arg-dn>
   <arg-value>
      <token-dest-dn/>
   </arg-value>
</do-remove-dest-attr-value>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-association	Association argument.
arg-value	Value argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA Class name of the object to modify. Can be required (for schema mapping purposes) if the object is other than the current object.	#IMPLIED
direct	true | false Use destCommandProcessor to carry out this action. Deprecated. Use when="direct" instead.	false
disabled	true | false True if this element is disabled.	false
name	CDATA Name of the attribute.	#REQUIRED

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false
when	auto | before | after | direct When this action should be performed: auto - Automatically determined (either in or after the current operation). before - Before the current operation. after - After the current operation. direct - Written directly to the destination data store instead of being added to the current document.	auto

Content Rule

((arg-dn | arg-association) ? , arg-value)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-remove-role

Initiates a request to the Roles Based Provisioning Module (RBPM) to revoke the Role specified by role-id from an Identity.

Remarks

The target Identity is specified by either <arg-dn> or <arg-association> if specified or by the current object otherwise. If specified by <arg-dn>, the DN must in LDAP format. If the target identity is specified by either <arg-dn> or <arg-association>, then the role-assignment-type must be specifed from one of USER_TO_ROLE, GROUP_TO_ROLE, CONTAINER_TO_ROLE or ROLE_TO_ROLE. If the role-assignment-type is not specified, then the assignment type is defaulted to USER_TO_ROLE.The request is made to the RBPM enabled User Application server specified by URL using credentials specified by id and <arg-password>. Additional optional arguments to the Role assignment request may be specified by the following named <arg-string>s.

Name	Description
role-assignment-type	The role assignment type from one of from one of USER_TO_ROLE, GROUP_TO_ROLE, CONTAINER_TO_ROLE or ROLE_TO_ROLE. Default: USER_TO_ROLE
description	A description of the reason for the request used for auditing and (if necessary) approval purposes. Default: Request generated by policy.
effective-time	The time (in CTIME format) the role assignment should become effective. Default: now

If any type of error occurs while requesting the role assignment, the error string is available to the enclosing policy in the local variable named *error.do-remove-rol*e. Otherwise that local variable is not available.

Example

```
<do-remove-role
    id="cn=RoleAdmin,o=People"
  url="http://localhost:8080/IDMProv"
role-
id="cn=Contractor,cn=Level30,cn=RoleDefs,cn=RoleConfig,cn=AppConfig,cn=UserApplica
tion,cn=DriverSet,o=novell"
  time-out="5000">
  <arg-password>
    <token-named-password name="role-admin"/>
  </arg-password>
  <arg-dn>
    <token-text xml:space="preserve">cn=Contractors,ou=Groups,o=Data</token-text>
  </arg-dn>
  <arg-string name="role-assignment-type">
    <token-text>GROUP_TO_ROLE</token-text>
  </arg-string>

  <arg-string name="description">
    <token-text>Requested by policy because isContractor set to false</token-text>
  </arg-string>
</do-remove-role>
```

Allowed Content

Element	Description
arg-password	Password argument.
arg-dn	DN argument.
arg-association	Association argument.
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
id	CDATA The LDAP format DN of a user authorized to make the request. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false
role-id	CDATA The LDAP format DN of the Role to revoke. Supports variable expansion.	#REQUIRED
time-out	CDATA The number of milliseconds you want Identity Manager to try to establish a connection to the User Application server before timing out. Supports variable expansion.	0
url	CDATA The URL of the User Application server hosting the Roles Based Provisioning Module. Supports variable expansion.	#REQUIRED

Content Rule

(arg-password, (arg-dn | arg-association) ? , arg-string *)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-remove-resource

Initiates a request to the Roles Based Provisioning Module (RBPM) to revoke the Resource specified by resource-id from an Identity.

Remarks

The target Identity is specified by either <arg-dn> or <arg-association> if specified or by the current object otherwise. If specified by <arg-dn>, the DN must be in the LDAP format. The request is made to the RBPM enabled User Application server specified by url using credentials specified by id and <arg-password>. Additional optional arguments to the Resource assignment request might be specified by the following named <arg-string>s.

Name	Description
description	A description of the reason for the request used for auditing and (if necessary) approval purposes. Default: Request generated by policy.

If any type of error occurs while requesting the resource assignment, the error string is available to the enclosing policy in the local variable named *error.do-remove-resource*. Otherwise that local variable is not available.

Example

```
<do-remove-resource
    id="CN=UAAdmin,OU=Sa,O=Data"
  url="http://localhost:8080/IDMProv"
  resource-
id="CN=Computer,CN=ResourceDefs,CN=RoleConfig,CN=AppConfig,CN=UserApplication,CN=D
riverSet,O=System"
  time-out="5000">
  <arg-password>
    <token-named-password name="resource-admin"/>
  </arg-password>
  <arg-string name="description">
    <token-text>Requested by policy because requireComputer set to false</token-
text>
  </arg-string>
</do-remove-resource>
```

Allowed Content

Element	Description
arg-password	Password argument.
arg-dn	DN argument.
arg-association	Association argument.
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
id	CDATA The LDAP format DN of a user authorized to make the request. Supports variable expansion.	#REQUIRED
instance-guid	CDATA The instance guid for revoking a single instance of a multivalued resource assignment. If null is specified, all instances are revoked. Supports variable expansion.	null
notrace	true | false True if this element should not be traced during execution of the policy.	false
requester-id	CDATA The LDAP format DN of a user who makes the request. Supports variable expansion.	#REQUIRED
resource-id	CDATA The LDAP format DN of the Resource to revoke. Supports variable expansion.	#REQUIRED
time-out	CDATA The number of milliseconds you want Identity Manager to try to establish a connection to the User Application server before timing out. Supports variable expansion.	0
url	CDATA The URL of the User Application server hosting the Roles Based Provisioning Module. Supports variable expansion.	#REQUIRED

Content Rule

(arg-password, (arg-dn | arg-association) ? , arg-string *)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Association argument.

do-remove-src-attr-value

Removes the value specified by <arg-value> from the named attribute on an object in the source data store. The target object is specified by either <arg-dn> or <arg-association> if specified, or by the current object if not specified.

Example

```
<do-remove-src-attr-value name="Member">
   <arg-dn>
      <token-text>Users/ManagerGroup</token-text>
   </arg-dn>
   <arg-value>
      <token-src-dn/>
   </arg-value>
</do-remove-src-attr-value>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-association	Association argument.
arg-value	Value argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA Class name of the object to modify. Can be required (for schema mapping purposes) if the object is other than the current object.	#IMPLIED
disabled	true | false True if this element is disabled.	false
name	CDATA Name of the attribute.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

((arg-dn | arg-association) ? , arg-value)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-rename-dest-object

Renames an object in the destination data store to the name specified by <arg-string>. The target object is specified by either <arg-dn> or <arg-association> if specified or by the current object otherwise.

Example

```
<do-rename-dest-object>
   <arg-dn>
      <token-text>Users/Active/Fred Flintstone</token-text>
   </arg-dn>
   <arg-string>
      <token-text>Fat Freddy</token-text>
   </arg-string>
</do-rename-dest-object>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-association	Association argument.
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA Class name of target object. Supports variable expansion.	#IMPLIED
direct	true | false Use destCommandProcessor to carry out this action. Deprecated. Use when="direct" instead.	false
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Attribute	Possible Values	Default Value
when	auto | before | after | direct When this action should be performed: auto - Automatically determined (either in or after the current operation). before - Before the current operation. after - After the current operation. direct - Written directly to the destination data store instead of being added to the current document.	auto

Content Rule

((arg-dn | arg-association) ? , arg-string)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-rename-op-attr

Causes all elements that are children of the current operation with the attr-name attribute equal to the name specified by src-name to have attr-name set to dest-name.

Example

```
<do-rename-op-attr src-name="Surname" dest-name="sn"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
dest-name	CDATA	#REQUIRED
	The new attribute name.	
disabled	true | false	false
	True if this element is disabled.	
notrace	true | false	false
	True if this element should not be traced during execution of the policy.	
src-name	CDATA	#REQUIRED
	The original attribute name.	

Content Rule

Empty

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-rename-src-object

Renames an object in the source data store to the name specified by <arg-string>. The target object is specified by either <arg-dn> or <arg-association> if specified, or by the current object if not specified.

Example

```
<do-rename-src-object>
   <arg-dn>
      <token-text>Users/Active/Fred Flintstone</token-text>
   </arg-dn>
   <arg-string>
      <token-text>Fat Freddy</token-text>
   </arg-string>
</do-rename-src-object>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-association	Association argument.
arg-value	Value argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA	#IMPLIED
	Class name of target object.	
	Support variable expansion.	
disabled	true | false	false
	True if this element is disabled.	
notrace	true | false	false
	True if this element should not be traced during execution of the policy.	

Content Rule

((arg-dn | arg-association) ? , arg-string)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-send-email

Sends an e-mail notification to the specified server. Optional credentials for authentication to the SMTP server are provided in the ID and password attributes. The type attribute identifies if the e-mail message contains plain text or HTML data. The various e-mail addresses, subject, and message are provided within <arg-string> elements and corresponding tag name attributes.

Remarks

Tag Name	Description
to	Adds the address to the list of e-mail recipients. Multiple instances are allowed.
cc	Adds the address to the list of CC e-mail recipients. Multiple instances are allowed.
bcc	Adds the address to the list of BCC e-mail recipients. Multiple instances are allowed.
from	Specifies the address to be used as the originating e-mail address.
reply-to	Specifies the address to be used as the e-mail message reply address.
subject	Specifies the e-mail subject.
message	Specifies the content of the e-mail message.
encoding	Specifies the character encoding to use for the e-mail message.
custom-smtp-header	Specifies a custom SMTP header to add to the email message.

Example

```
<do-send-email server="smtp.company.com" id="user" password="emailpwd"
type="text">
   <arg-string name="to">
      <token-text>to_user1@company.com</token-text>
   </arg-string>
   <arg-string name="to">
      <token-text>to_user2@company.com</token-text>
   </arg-string>
   <arg-string name="cc">
      <token-text>cc_user@company.com</token-text>
   </arg-string>
   <arg-string name="bcc">
      <token-text>bcc_user@company.com</token-text>
   </arg-string>
   <arg-string name="from">
      <token-text>from_user@company.com</token-text>
   </arg-string>
   <arg-string name="subject">
      <token-text>This is the email subject</token-text>
   </arg-string>
   <arg-string name="message">
      <token-text>This is the email body</token-text>
   </arg-string>
</do-send-email>
```

Allowed Content

Element	Description
arg-string	String argument.
arg-password	Password argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
id	CDATA User account on the SMTP server. Support variable expansion.	#IMPLIED
password	CDATA Password for the user on the SMTP server. Deprecated; use <arg-password> with <token-named-password> instead.	#IMPLIED
server	CDATA DNS name or IP address of the SMTP server. Support variable expansion.	#REQUIRED

Attribute	Possible Values	Default Value
type	text | html Identifies if e-mail message contains plain text or HTML data.	text

Content Rule

(arg-string *)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-send-email-from-template

Generates an e-mail notification using an SMTP notification configuration object, e-mail template object, and replacement tokens. The target SMTP server along with credentials for authentication and the originating address are read from the SMTP notification configuration object. The subject and e-mail message are created using the template object and template replacement tokens. Replacement tokens are declared within a <arg-string> element and tag name attribute. The value of <arg-string> tag attribute is interpretted as html, if it is enclosed within <use-html></use-html> tags.Reserved replacement tokens specify the various recipient addresses.

Remarks

Reserved Token	Description
to	Adds the address to the list of e-mail recipients. Multiple instances are allowed.
cc	Adds the address to the list of CC e-mail recipients. Multiple instances are allowed.
bcc	Adds the address to the list of BCC e-mail recipients. Multiple instances are allowed.
reply-to	Specifies the address to be used as the e-mail message reply address.
encoding	Specifies the character encoding to use for the e-mail message.

Example

```
<do-send-email-from-template
     notification-dn="/cn=security/cn=DefaultNotification Collection"
     template-dn="/cn=security/cn=DefaultNotification Collection/cn=PS
     Sync Fail">
   <arg-password>
      <token-named-password name="email-server"/>
   </arg-password>
   <arg-string name="manager">
      <token-text>Bill Jones</token-text>
   </arg-string>
   <arg-string name="surname">
      <token-text>Smith</token-text>
   </arg-string>
   <arg-string name="given-name">
      <token-text>Joe</token-text>
   </arg-string>
   <arg-string name="to">
```

```
      <token-text>to_user@company.com</token-text>
   </arg-string>
   <arg-string name="cc">
      <token-text>cc_user@company.com</token-text>
   </arg-string>
   <arg-string name="custom-smtp-header">
      <token-text>X-Priority: 1(Highest)</token-text>
   </arg-string>
   <arg-string name="FailureReason">
      <token-text>
         <use-html><p>sample reason 1</p><p>sample reason 2</p></use-html>
      </token-text>
   </arg-string>
</do-send-email-from-template>
```

Allowed Content

Element	Description
arg-password	Password argument.
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
notification-dn	CDATA Slash form DN of the SMTP notification configuration object.	#REQUIRED
disabled	true | false True if this element is disabled.	false
password	CDATA Password for user on the SMTP server. Deprecated; use <arg-password> with <token-named-password> instead.	#IMPLIED
template-dn	CDATA Slash form DN of the e-mail template object. Supports variable expansion.	#REQUIRED

Content Rule

(arg-string *)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-set-default-attr-value

Adds the values specified by <arg-value> to the current operation for named attribute if no values for that attribute already exist. It is only valid when the current operation is <add>. If write-back=true default values are also written back to the source object.

Example

```
<do-set-default-attr-value name="L">
   <arg-value>
      <token-text>Unknown</token-text>
   </arg-value>
</do-set-default-attr-value>
```

Allowed Content

Element	Description
arg-value	Value argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
name	CDATA Name of the attribute. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false
write-back	true | false True if the default value should also be written back to the source object.	false

Content Rule

(arg-value +)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-set-dest-attr-value

Adds the value specified by <arg-value> to the named attribute on an object in the destination data store, and removes all other values for that attribute. The target object is specified by either <arg-dn> or <arg-association> if specified, or by the current object if not specified.

Example

```
<do-set-dest-attr-value name="OU">
   <arg-value>
      <token-text>Sales</token-text>
   </arg-value>
</do-set-dest-attr-value>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-association	Association argument.
arg-value	Value argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA Class name of object to modify. Can be required (for schema mapping purposes) if the object is other than the current object.	#IMPLIED
direct	true | false Use destCommandProcessor to carry out this action. Deprecated. Use when="direct" instead.	false
disabled	true | false True if this element is disabled	false
name	CDATA Name of the attribute.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Attribute	Possible Values	Default Value
when	auto | before | after | direct When this action should be performed: auto - Automatically determined (either in or after the current operation). before - Before the current operation. after - After the current operation. direct - Written directly to the destination data store instead of being added to the current document.	auto

Content Rule

((arg-dn | arg-association) ? , arg-value)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-set-dest-password

Sets the value specified by <arg-string>'s as the old password (optional) and new password for the current object in the destination data store. The target object is specified by <arg-dn> or <arg-association> if specified or by the current object otherwise.

Example

```
<do-set-dest-password>
  <arg-dn>
    <token-text>Users/Fred Flintstone</token-text>
  </arg-dn>
  <arg-string>
    <token-text>oldpassword</token-text>
  </arg-string>
  <arg-string>
    <token-text>newpassword</token-text>
  </arg-string>
</do-set-dest-password>
```

IMPORTANT: When specifying both the old password and the new password, the old password must be specified in the first arg-string.

Allowed Content

Element	Description
arg-dn	DN argument.
arg-association	Association argument.
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA	#IMPLIED
	Class name of target object.	
	Supports variable expansion.	
direct	true | false	false
	Use destCommandProcessor to carry out this action.	
	Deprecated. Use when="direct" instead.	
disabled	true | false	false
	True if this element is disabled.	

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false
when	auto | before | after | direct When this action should be performed: auto - Automatically determined (either in or after the current operation). before - Before the current operation. after - After the current operation. direct - Written directly to the destination data store instead of being added to the current document.	auto

Content Rule

((arg-dn | arg-association) ? , arg-string, arg-string ?)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-set-local-variable

Sets a local variable with the given name to the string value specified by <arg-string>, the XPath 1.0 node set specified by <arg-node-set>, or the Java* object specified by <arg-object>.

Example

```
<!-- Sets variable to the string value of the first value of Surname -->
<do-set-local-variable name="lastName">
   <arg-string>
      <token-attr name="Surname"/>
   </arg-string>
</do-set-local-variable>

<!-- sets variable equal to all the value elements of Surname -->
<do-set-local-variable name="lastName">
   <arg-node-set>
      <token-attr name="Surname"/>
   </arg-node-set>
</do-set-local-variable>

<!-- sets variable equal to an instance of java.util.Random -->
<!-- note that the prefix jrandom needs to have been mapped to -->
<!-- the URI http://www.novell.com/nxsl/java/java.util.Random -->
<!-- on the <policy> -->
<do-set-local-variable name="lastName">
   <arg-object>
      <token-xpath expression="jrandom:new()"/>
   </arg-object>
</do-set-local-variable>
```

Allowed Content

Element	Description
arg-string	String argument.
arg-node-set	Node set argument.
arg-object	Java object argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false	false
	True if this element is disabled.	
name	NMTOKEN	#REQUIRED
	Name of the variable. Supports variable expansion.	

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false
scope	policy | driver Scope of the variable. Policy: Variable is visible only within the current policy during the current invocation of the policy. Driver: Variable is visible to all policies within the current driver until the driver is stopped. Supports variable expansion.	policy

Content Rule

(arg-string | arg-node-set | arg-object)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-set-op-association

Sets the association value for the current operation to the value provided by <arg-association>.

Example

```
<do-set-op-association>
   <arg-association>
      <token-src-name/>
   </arg-association>
</do-set-op-association>
```

Allowed Content

Element	Description
arg-association	Association argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(arg-association)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-set-op-class-name

Causes the object class name for the current operation to be set to the value provided by <arg-string>.

Example

```
<do-set-op-class-name>
   <arg-string>
      <token-text>User</token-text>
      <token-src-name/>
   </arg-string>
</do-set-op-class-name>
```

Allowed Content

Element	Description
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(arg-string)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-set-op-dest-dn

Sets the destination DN for the current operation to the value provided by <arg-dn>.

Example

```
<do-set-op-dest-dn>
   <arg-dn>
      <token-text>Novell\Users\</token-text>
   </arg-dn>
</do-set-op-dest-dn>
```

Allowed Content

Element	Description
arg-dn	DN argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(arg-dn)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-set-op-property

Sets an operation property with the given name to the value specified by <arg-string> on the current operation. An operation property is a named value that is stored as an attribute on an <operation-data> element within an operation and is typically used to supply additional context that might be needed by the policy that handles the results of an operation.

Example

```
<do-set-op-property name="myProperty">
   <arg-string>
      <token-text>Fred</token-text>
   </arg-string>
</do-set-op-property>
```

Allowed Content

Element	Description
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false	false
	True if this element is disabled.	
name	NMTOKEN	#REQUIRED
	Name of the operation property.	
notrace	true | false	false
	True if this element should not be traced during execution of the policy.	

Content Rule

(arg-string)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-set-op-src-dn

Sets the source DN for the current operation to the value provided by <arg-dn>.

Example

```
<do-set-op-src-dn>
 <arg-dn>
 <token-text>Novell\Users\</token-text>
 <token-attr name="CN"/>
 </arg-dn>
</do-set-op-src-dn>
```

Allowed Content

Element	Description
arg-dn	DN argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(arg-dn)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-set-op-template-dn

Sets the template DN for the current operation to the value provided by <arg-dn>. It is only valid when the current operation is <add>.

Example

```
<do-set-op-template-dn>
   <arg-dn>
      <token-text>Novell\Users\UserTemplate</token-text>
   </arg-dn>
</do-set-op-template-dn>
```

Allowed Content

Element	Description
arg-dn	DN argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(arg-dn)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-set-src-attr-value

Adds the value specified by <arg-value> to the named attribute on an object in the source data store and all other values for that attribute are removed. The target object is specified by either <arg-dn> or <arg-association> if specified or by the current object otherwise.

Example

```
<do-set-src-attr-value name="OU">
   <arg-value>
      <token-text>Sales</token-text>
   </arg-value>
</do-set-src-attr-value>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-association	Association argument.
arg-value	Value argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA Class name of object to modify. Can be required (for schema mapping purposes) if object is other than the current object.	#IMPLIED
disabled	true | false True if this element is disabled.	false
name	CDATA Name of the attribute.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

((arg-dn | arg-association) ? , arg-value)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-set-src-password

Sets the value specified by <arg-string>s as the old password (optional) and new password for the current object in the destination data store. The target object is specified by either <arg-dn> or <arg-association> if specified or by the current object otherwise.

Example

```
<do-set-src-password>
  <arg-dn>
    <token-text>Users/Fred Flintstone</token-text>
  </arg-dn>
  <arg-string>
    <token-text>oldpassword</token-text>
  </arg-string>
  <arg-string>
    <token-text>newpassword</token-text>
  </arg-string>
</do-set-src-password>
```

IMPORTANT: When specifying both the old password and the new password, the old password must be specified in the first arg-string.

Allowed Content

Element	Description
arg-dn	DN argument.
arg-association	Association argument.
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA Class name of target object. Supports variable expansion.	#IMPLIED
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

((arg-dn | arg-association) ? , arg-string, arg-string ?)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-set-sso-credential

Sets a credential on the object specified by the <arg-dn> element in the Single Sign On credential store specified by the <store-def-dn> element for the application specified by the <app-id> element. The credential information is specified by additional named <arg-string> elements. The number of the strings and the names used are dependent on the credential store and application for which the credential is targeted.

Example

```
<do-set-sso-credential store-def-dn="../Library/SSO1" app-id="AD7">
   <arg-dn>
      <token-parse-dn src-dn-format="src-dn" dest-dn-format="ldap" start="0"
       length="-1">
         <token-src-dn/>
      </token-parse-dn>
   </arg-dn>
   <arg-string name="username">
      <token-src-name/>
   </arg-string>
   <arg-string name="password">
      <token-local-variable name="generatedPassword"/>
   </arg-string>
</do-set-sso-credential>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
app-def-dn	CDATA DN of the application credential definition object Only used by the UI so the various UIs should agree on the DN format used.	#IMPLIED
app-id	CDATA Application ID for the credential.	#REQUIRED
disabled	true | false True if this element is disabled.	false

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false
store-def-dn	CDATA Slash form DN of the credential store definition object. Can be relative to the including policy.	#REQUIRED

Content Rule

(arg-dn , arg-string *)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-set-sso-passphrase

Sets the passphrase question and answer specified by <arg-string> elements on the object specified by the <arg-dn> element in the Single Sign On credential store specified by the <store-def-dn> element.

Example

```
<do-set-sso-passphrase store-def-dn="../Library/SSO1">
   <arg-dn>
      <token-parse-dn src-dn-format="src-dn" dest-dn-format="ldap" start="0"
       length="-1">
         <token-src-dn/>
      </token-parse-dn>
   </arg-dn>
   <arg-string>
      <token-text/>What favorite color?<token-text/>
   </arg-string>
   <arg-string>
      <token-text/>blue<token-text/>
   </arg-string>
</do-set-sso-passphrase>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false
store-def-dn	CDATA {description of store-def-dn}	#REQUIRED

Content Rule

(arg-dn , arg-string , arg-string)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-set-xml-attr

Sets a custom XML attribute named by the name attribute on the set of elements selected by expression.

Example

```
<do-set-xml-attr name="cert-id" expression=".">
   <arg-string>
      <token-text>c:\lotus\domino\data\eng.id</token-text>
   </arg-string>
</do-set-xml-attr>
<do-set-xml-attr name="cert-pwd" expression=".">
   <arg-string>
      <token-text>certify2eng</token-text>
   </arg-string>
</do-set-xml-attr>
```

Allowed Content

Element	Description
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
expression	CDATA XPath 1.0 expression that returns a node-set containing the elements on which the XML attribute should be set.	#REQUIRED
name NMTOKEN	Tag name of the XML attribute. Might contain a namespace prefix if that prefix has been defined on the <policy>. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(arg-string)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-start-workflow

Starts the workflow specified by workflow-id for the recipient specified by <arg-dn> on the User Application server specified by the URL using credentials specified by ID and <arg-password>. The recipient must be an LDAP format DN of an object in the directory served by the User Application server. The additional arguments to the workflow may be specified by named <arg-string>. You can specify multiple values and delimit them by using a semi-colon(;). Use a backslash (\) to escape the semi-colon(;). The number of the strings and the names used depend on the workflow to be started. There are some names that have special meaning and are available regardless of the workflow being started.

Remark

- **:InitiatorOverrideDN:** The LDAP format DN of the initiator of the workflow, if other than the user used to authenticate.
- **:CorrelationID:** An identifier used to correlate related workflows.

If any type of error occurs while starting the workflow, the error string will be available to the enclosing policy in the local variable named error.do-start-workflow. Otherwise that local variable will be unavailable.

Example

```
<do-start-workflow id="cn=WorkflowAdmin,o=People" url="http://localhost:8080/
IDMProv" workflow-
id="CN=ApproveCellPhone,CN=RequestDefs,CN=AppConfig,CN=UserApplication,CN=DriverSe
t,O=novell" time-out="5000">
  <arg-password>
    <token-named-password name="workflow-admin"/>
  </arg-password>
  <arg-dn>
    <token-parse-dn dest-dn-format="ldap" src-dn-format="qualified-slash">
      <token-xpath expression="@qualified-src-dn"/>
    </token-parse-dn>
  </arg-dn>
  <arg-string name="provider">
    <token-text>ACMEWireless</token-text>
  </arg-string>
  <arg-string name="reason">
    <token-text>new hire</token-text>
  </arg-string>
  <arg-string name="email">
    <token-text>jmiller@acme.com; jack.miller@gmail.com; jackm@outlook.com</token-
text>
  </arg-string>
  <arg-string name="text">
    <token-text>one, two, and three\; a, b, and c\; first, second, and third</
token-text>
  </arg-string>
</do-start-workflow>
```

Allowed Content

Element	Description
arg-password	Password argument.
arg-dn	DN argument.
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True, if this element is disabled.	false
id	CDATA The LDAP format DN of a user authorized to start workflows on the User Application server. Supports variable expansion.	#REQUIRED
notrace	true | false True, if this element should not be traced during execution of the policy.	false
time-out	CDATA The number of milliseconds Identity Manager should wait to establish a connection with the User Application server before timing out. Supports variable expansion.	0
url	CDATA The URL of the User Application server where the workflow runs. Supports variable expansion.	#REQUIRED
workflow-id	CDATA The LDAP format DN of the workflow to start. Supports variable expansion.	#REQUIRED

Content Rule

(arg-password , arg-dn, arg-string *)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-status

Generates a status notification with the specified level and with a message provided by <arg-string>.

Remarks

If the level is retry, then the policy immediately halts processing of the input document and schedule a retry of the event currently being processed.

If the level is fatal, then the policy immediately halts processing of the input document and initiates a shutdown of the driver.

If a the current operation is an event-id, then that event-id is used for the status notification; otherwise, there is no event-id reported.

Example

```
<do-status level="warning">
   <arg-string>
      <token-src-dn/>
      <token-text>: operation vetoed on out-of-scope object</token-text>
   </arg-string>
</do-status >
```

Allowed Content

Element	Description
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
level	CDATA Status level.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(arg-string)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-strip-op-attr

Strips all elements that are children of the current operation with the attr-name attribute equal to the name specified by name from the current operation.

Example

```
<do-strip-op-attr name="Member"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
name	CDATA Name of the attribute.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

Empty

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-strip-xpath

This action can remove nodes selected by the XPath 1.0 expression from the current operation or from any node set type variable. The expression must evaluate to a node set.

Example

```
<do-strip-xpath expression="*[@attr-name='OU']"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
disabled	true | false	false
	True if this element is disabled.	
expression	CDATA	#REQUIRED
	XPath expression.	
notrace	true | false	false
	True if this element should not be traced during execution of the policy.	

Content Declaration

Empty

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-trace-message

Causes the trace message provided by <arg-string> to DSTRACE if the specified level is less than or equal to the currently configured trace level.

Example

```
<do-trace-level level="0" color="blue">
   <arg-string>
      <token-text>placing new object at </token-text>
      <token-dest-dn/>
   </arg-string>
</do-status >
```

Allowed Content

Element	Description
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
color	black | blue | green | cyan | red | purple | brown | grey | drgrey | brblue | brgreen | brcyan | brred | brpurple | yellow | white Color of the text to send.	brpurple
disabled	true | false True if this element is disabled.	false
level	CDATA Minimum trace level at which to send the message.	0
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(arg-string)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-veto

Cancels the current operation.

Example

```
<do-veto/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

Empty

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-veto-if-op-attr-not-available

Cancels the current operation if the named attribute is not available in the current operation.

Example

```
<do-veto-if-op-attr-not-available name="CN"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
name	CDATA Name of the attribute. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

Empty

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

do-while

Causes the actions specified by <arg-actions> to be repeated while the conditions specified by <arg-conditions> evaluate to true.

Example

```
<do-set-local-variable name="counter">
   <arg-string>
      <token-text>1</token-text>
   </arg-string>
</do-set-local-variable>
<do-while>
   <arg-conditions>
      <and>
         <if-local-variable name="counter" op="not-gt" mode="numeric">10</if-local-
variable>
      </and>
   </arg-conditions>
   <arg-actions>
      <do-trace-message level="0" color="yellow">
         <arg-string>
            <token-text>Counter = </token-text>
            <token-local-variable name="counter"/>
         </arg-string>
      </do-trace-message>
      <do-set-local-variable name="counter">
         <arg-string>
            <token-xpath expression="$counter + 1"/>
         </arg-string>
      </do-set-local-variable>
   </arg-actions>
</do-while>
```

Allowed Content

Element	Description
arg-conditions	Conditions argument.
arg-actions	Actions argument.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(arg-conditions , arg-actions)

Parent Elements

Element	Description
actions	Actions that are performed by a <rule>.
arg-actions	Actions argument.

if-association

Performs a test on the association value of the current operation or the current object. The type of test performed depends on the operator specified by the op attribute. The following table shows the type of test performed by each operator.

Remarks

Operator	Returns true when...
associated	There is an established association for the current object.
available	There is a non-empty association value specified by the current operation.
equal	The association value specified by the current operation is exactly equal to the content of <if-association>. Supports variable expansion.
lt	The association value specified by the current operation is less than the content of the condition when compared using the specified comparison mode. Supports variable expansion.
gt	The association value specified by the current operation is greater than the content of the condition when compared using the specified comparison mode. Supports variable expansion.
not-associated	Associated returns false.
not-available	Available returns false.
not-equal	Equal returns false.
not-lt	Less than returns false.
not-gt	Greater than returns false.

Example

```
<if-association op="associated "/>
<if-association op="available"/>
<if-association
op="equal">{07414faa-1b38-40ec-8b7c-c20aa21ddafb}</if-association>
```

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
mode	case | nocase | regex | src-dn | dest-dn | numeric | octet Comparison mode if op implies a comparison.	nocase
notrace	true | false True if this element should not be traced during execution of the policy.	false
op	available | equal | lt | gt | not-available | not-equal not-lt | not-gt Test operator.	#REQUIRED

Content Rule

(#PCDATA)

Parent Elements

Element	Description
and	Logical conjunction.
or	Logical disjunction.

if-attr

Performs a test on attribute values of the current object in either the current operation or the source data store.

Remarks

It can logically be thought of as equivalent to:

```
<or>
 <if-op-attr/>
 <if-src-attr/>
</or>
```

Operator	Returns true when...
available	There is a value available in either the current operation or the source data store for the specified attribute.
equal	There is a value available in either the current operation or the source data store for the specified attribute that equals the content of <if-attr> when compared using the specified comparison mode. If mode="structured" then the content must be a set of <component> elements otherwise, it must be text. Supports variable expansion.
lt	There is a value available in either the current operation or the source data store for the specified attribute that is less than the content of the condition when compared using the specified comparison mode. Supports variable expansion.
gt	There is a value available in either the current operation or the source data store for the specified attribute that is greater than the content of the condition when compared using the specified comparison mode. Supports variable expansion.
not-available	Available returns false.
not-equal	Equal returns false.
not-lt	Less than returns false.
not-gt	Greater than returns false.

Example

```
<if-attr op="available" name="OU"/>
<if-attr op="equal" mode="nocase" name="OU">Sales</if-attr>
<if-attr op="equal" mode="structured" name="Language">
 <component name="string">EN</component>
 <component name="string">JP</component>
</if-attr >
```

Allowed Content

#PCDATA

Element	Description
component	Value component.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false	false
	True if this element is disabled.	
mode	case | nocase | regex | src-dn | dest-dn | numeric | octet | structured	nocase
	Comparison mode if op implies a comparison.	
name	CDATA	#REQUIRED
	Name of the attribute.	
notrace	true | false	false
	True if this element should not be traced during execution of the policy.	
op	available | equal | lt | gt | not-available | not-equal | not-lt | not-gt	#REQUIRED
	Test operator.	

Content Rule

(#PCDATA | component) *

Parent Elements

Element	Description
and	Logical conjunction.
or	Logical disjunction.

if-class-name

Performs a test on the object class name in the current operation.

Remarks

Operator	Returns true when...
available	There is an object class name available in the current operation.
equal	There is an object class name available in the current operation and it equals the content of <if-class-name> when compared using the specified comparison mode.
lt	There is an object class name available in the current operation and it is less than the content of the condition when compared using the specified comparison mode. Supports variable expansion.
gt	There is an object class name available in the current operation and it is greater than the content of the condition when compared using the specified comparison mode. Supports variable expansion.
not-available	Available returns false.
not-equal	Equal returns false.
not-lt	Less than returns false.
not-gt	Greater than returns false.

Example

```
<if-class-name op="available"/>
<if-class-name op="equal" mode="nocase">User</if-class-name >
```

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
mode	case | nocase | regex | src-dn | dest-dn | numeric | octet Comparison mode if op implies a comparison.	nocase
notrace	true | false True if this element should not be traced during execution of the policy.	false
op	available | equal | lt | gt | not-available | not-equal not-lt | not-gt Test operator.	#REQUIRED

Content Rule

(#PCDATA)

Parent Elements

Element	Description
and	Logical conjunction.
or	Logical disjunction.

if-dest-attr

Performs a test on attribute values of the current object in the destination data store. The type of test performed depends on the operator specified by the op attribute. The table below shows the type of test performed by each operator.

Remarks

Operator	Returns true when...
available	There is a value available in the destination data store for the specified attribute.
equal	There is a value available for the specified attribute in the destination data store that equals the content of <if-dest-attr> when compared using the specified comparison mode. If mode="structured" then the content must be a set of <component> elements otherwise, it must be text. Supports variable expansion.
lt	There is a value available for the specified attribute in the destination data store that is less than the content of the condition when compared using the specified comparison mode. If mode="structured" then the content must be a set of <component> elements, otherwise it must be text. Supports variable expansion.
gt	There is a value available for the specified attribute in the destination data store that is greater than the content of the condition when compared using the specified comparison mode. If mode="structured" then the content must be a set of <component> elements, otherwise it must be text. Supports variable expansion.
not-associated	Associated returns false.
not-available	Available returns false.
not-equal	Equal returns false.
not-lt	Less than returns false.
not-gt	Greater than returns false.

Example

```
<if-dest-attr op="available" name="OU"/>
<if-dest-attr op="equal" mode="nocase" name="OU">Sales</if-dest-attr>
<if-dest-attr op="equal" mode="structured" name="Language">
 <component name="string">EN</component>
 <component name="string">JP</component>
</if-dest-attr >
```

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
mode	case | nocase | regex | src-dn | dest-dn | numeric | octet | structured Comparison mode if op="equal" or op="not-equal"	nocase
name	CDATA Name of the attribute. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false
op	available | equal | lt | gt | not-available | not-equal | not-lt | not-gt Test operator.	#REQUIRED

Content Rule

(#PCDATA | component) *

Parent Elements

Element	Description
and	Logical conjunction.
or	Logical disjunction.

if-dest-dn

Performs a test on the destination DN in the current operation.

Remarks

Operator	Returns true when...
available	There is a destination DN available.
equal	There is a destination DN available and it equals the content of <if-dest-dn> when compared using semantics appropriate to the DN format of the destination data store. Supports variable expansion.
in-container	There is a destination DN available and it represents an object in the container specified by the content of <if-dest-dn> when compared using semantics appropriate to the DN format of the destination data store. Supports variable expansion.
in-subtree	There is a destination DN available and it represents and object in the subtree specified by the content of <if-dest-dn> when compared using semantics appropriate to the DN format of the destination data store. Supports variable expansion.
not-available	Available returns false.
not-equal	Equal returns false.
not-lt	Less than returns false.
not-gt	Greater than returns false.

Example

```
<if-dest-dn op="available"/>
<if-dest-dn op="equal">Novell\Users\Fred</if-dest-dn>
<if-dest-dn op="in-container">Novell\Users</if-dest-dn>
<if-dest-dn op="in-subtree">Novell</if-dest-dn >
```

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
Attribute	disabled true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false
op	available | equal | in-container | in-subtree | not-available | not-equal | not-in-container | not-in-subtree Test operator.	#REQUIRED

Content Rule

(#PCDATA)

Parent Elements

Element	Description
and	Logical conjunction.
or	Logical disjunction.

if-entitlement

Performs a test on entitlements of the current object in either the current operation or the Identity Vault data store.

Remarks

Operator	Returns true when...
available	The named entitlement is available and granted in either the current operation or the Identity Vault data store.
changing	The current operation contains a change (grant or revoke) of the named entitlement.
changing-from	The current operation contains a change that revokes a value of the named entitlement that has a value that equals the content of <if-entitlement> when compared using the specified comparison mode. Supports variable expansion.
changing-to	The current operation contains a change that grants a value of the named entitlement that has a value that equals the content of <if-entitlement> when compared using the specified comparison mode. Supports variable expansion.
equal	The named entitlement is available and granted in either the current operation or the Identity Vault data store and has a value that equals the content of <if-entitlement> when compared using the specified comparison mode. Supports variable expansion.
lt	The named entitlement is available and granted in either the current operation or the Identity Vault data store and has a value that is less than the content of the condition when compared using the specified comparison mode. Supports variable expansion.
gt	The named entitlement is available and granted in either the current operation or the Identity Vault data store and has a value that is greater than the content of the condition when compared using the specified comparison mode. Supports variable expansion.
not-available	Available returns false.
not-changing	Changing returns false
not-changing-from	Changing-from returns false.
not-changing-to	Changing-to returns false.
not-equal	Equal returns false.
not-lt	Less than returns false.
not-gt	Greater than returns false.

Example

```
<if-entitlement op="available" name="notes-group"/>
<if-entitlement op="changing" name="notes-group"/>
<if-entitlement op="changing-from" name="notes-group"/>Sales</if-entitlement>
<if-entitlement op="changing-to" name="notes-group"/>Sales</if-entitlement>
<if-entitlement op="equal" mode="nocase" name="notes-group">Sales</if-entitlement>
```

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
disabled	true | false	false
	True if this element is disabled.	
mode	case | nocase | regex | src-dn | dest-dn | numeric | octet	nocase
	Comparison mode if op="equal" or op="not-equal" or op="changing-from" or op="changing-to".	
name	CDATA	#REQUIRED
	Name of the entitlement. Supports variable expansion.	
notrace	true | false	false
	True if this element should not be traced during execution of the policy.	
op	available | changing | changing-from | changing-to | equal | lt | gt | not-available | not-changing | not-changing-from | not-changing-to | not-equal | not-lt | not-gt	#REQUIRED
	Test operator.	

Content Rule

(#PCDATA)

Parent Elements

Element	Description
and	Logical conjunction.
or	Logical disjunction.

if-global-variable

Performs a test on a global configuration variable. The type of test performed depends on the operator specified by the op attribute. The following table shows the type of test performed by each operator.

Remarks

Operator	Returns true when...
available	There is a global configuration variable with the specified name.
equal	There is a global configuration variable with the specified name and its value equals the content of <if-global-variable> when compared using the specified comparison mode. Supports variable expansion.
lt	There is a global configuration variable with the specified name and its value is less than the content of the condition when compared using the specified comparison mode. Supports variable expansion.
gt	There is a global configuration variable with the specified name and its value is greater than the content of the condition when compared using the specified comparison mode. Supports variable expansion.
not-available	Available returns false.
not-equal	Equal returns false.
not-lt	Less than returns false.
not-gt	Greater than returns false.

Example

```
<if-global-variable op="available" name="myGlobalVariable"/>
<if-global-variable op="equal" mode="nocase" name="myGlobalVariable">enabled</if-
global-variable>
```

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false

Attribute	Possible Values	Default Value
mode	case | nocase | regex | src-dn | dest-dn | numeric | octet comparison mode if op implies a comparison.	nocase
name	CDATA Name of the variable. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false
op	available | equal | lt | gt | not-available | not-equal | not-lt | not-gt Test operator.	#REQUIRED

Content Rule

(#PCDATA)

Parent Elements

Element	Description
and	Logical conjunction.
or	Logical disjunction.

if-local-variable

Performs a test on a local variable. The type of test performed depends on the operator specified by the op attribute. The following table shows the type of test performed by each operator. If local variable holds a node set, the string value of each node in the node set is used for comparison purposes. If the same local variable exists in both the policy scope and the driver scope, the variable in the policy scope takes precedence.

Remarks

Operator	Returns true when...
available	There is a local variable with the specified name that has been defined by an action of an earlier <rule> within the <policy>.
equal	There is a local variable with the specified name and its value equals the content of <if-local-variable> when compared using the specified comparison mode. Supports variable expansion.
lt	There is a local variable with the specified name and its value is less than the content of the condition when compared using the specified comparison mode. Supports variable expansion.
gt	There is a local variable with the specified name and its value is greater than the content of the condition when compared using the specified comparison mode. Supports variable expansion.
not-available	Available returns false.
not-equal	Equal returns false.
not-lt	Less than returns false.
not-gt	Greater than returns false.

Example

```
<if-local-variable op="available" name="myLocalVariable"/>
<if-local-variable op="equal" mode="nocase" name="myLocalVariable">enabled</if-
local-variable>
```

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
mode	case | nocase | regex | src-dn | dest-dn | numeric | octet Comparison mode if op="equal" or op="not-equal".	nocase
name	CDATA Name of the variable. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false
op	available | equal | lt | gt | not-available | not-equal | not-lt | not-gt Test operator.	#REQUIRED

Content Rule

(#PCDATA)

Parent Elements

Element	Description
and	Logical conjunction.
or	Logical disjunction.

if-named-password

Performs a test on a named password from the driver. The type of test performed depends on the operator specified by the op attribute. The following table shows the type of test performed by each operator.

Remarks

Operator	Returns true when...
available	There is password with the specified name available.
not-available	Available returns false.

Example

```
<if-named-password op="available" name="extraPassword"/>
```

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
disabled	true | false	false
	True if this element is disabled.	
name	CDATA	#REQUIRED
	Name of the password.	
notrace	true | false	false
	True if this element should not be traced during execution of the policy.	
op	available | not-available	#REQUIRED
	Test operator.	

Content Declaration

(#PCDATA)

Parent Elements

Element	Description
and	Logical conjunction.
or	Logical disjunction.

if-op-attr

Performs a test on attribute values in the current operation.

Remarks

Operator	Returns true when...
available	There is a value available in the current operation (<add-attr>, <add-value> or <attr>) for the specified attribute.
changing	The current operation contains a change (<modify-attr> or <add-attr>) of the specified attribute.
changing-from	The current operation contains a change that removes a value (<remove-value>) of the specified attribute that equals the content of <if-op-attr> when compared using the specified comparison mode. If mode="structured", then the content must be a set of <component> elements; otherwise, it must be text.
changing-to	The current operation contains a change that adds a value (<add-value> or <add-attr>) to the specified attribute that equals the content of <if-op-attr> when compared using the specified comparison mode. If mode="structured", then the content must be text; otherwise, it must be a set of <component> elements.
equal	There is a value available in the current operation (other than a <remove-value>) for the specified attribute that equals the content of <if-op-attr> when compared using the specified comparison mode. If mode="structured", then the content must be a set of <component> elements; otherwise, it must be text. Supports variable expansion.
lt	There is a value available in the current operation (other than a <remove-value>) for the specified attribute that is less than the content of the condition when compared using the specified comparison mode. If mode="structured" then the content must be a set of <component> elements, otherwise it must be text. Supports variable expansion.
gt	There is a value available in the current operation (other than a <remove-value>) for the specified attribute that is greater than the content of the condition when compared using the specified comparison mode. If mode="structured" then the content must be a set of <component> elements, otherwise it must be text. Supports variable expansion.
not-available	Available returns false.
not-changing	Changing returns false
not-changing-from	Changing-from returns false.

Operator	Returns true when...
not-changing-to	Changing-to returns false.
not-equal	Equal returns false.
not-lt	Less than returns false.
not-gt	Greater than returns false.

Example

```
<if-op-attr op="available" name="OU"/>
<if-op-attr op="changing" name="OU"/>
<if-op-attr op="changing-from" name="OU"/>Sales</if-op-attr>
<if-op-attr op="changing-to" name="OU"/>Sales</if-op-attr>
<if-op-attr op="equal" mode="nocase" name="OU">Sales</if-op-attr>
<if-op-attr op="equal" mode="structured" name="Language">
 <component name="string">EN</component>
 <component name="string">JP</component>
</if-op-attr>
```

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
mode	case | nocase | regex | src-dn | dest-dn | numeric | octet | structured Comparison mode if op="equal" or op="not-equal" or op="changing-from" or op="changing-to".	nocase
name	CDATA Name of the attribute. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false
op	available | changing | changing-from | changing-to | equal | lt | gt | not-available | not-changing | not-changing-from | not-changing-to | not-equal | not-lt | not-gt Test operator.	#REQUIRED

Content Rule

(#PCDATA | component) *

Parent Elements

Element	Description
and	Logical conjunction.
or	Logical disjunction.

if-op-property

Performs a test on an operation property on the current operation. The type of test performed depends on the operator specified by the op attribute. The following table shows the type of test performed by each operator.

Remarks

Operator	Returns true when...
available	There is an operation property with the specified name on the current operation.
equal	There is a an operation property with the specified name on the current operation and its value equals the content of <if-op-property> when compared using the specified comparison mode. Supports variable expansion.
lt	There is a an operation property with the specified name on the current operation and its value is less than the content of the condition when compared using the specified comparison mode. Supports variable expansion.
gt	There is a an operation property with the specified name on the current operation and its value is greater than the content of the condition when compared using the specified comparison mode. Supports variable expansion.
not-available	Available returns false.
not-equal	Equal returns false.
not-lt	Less than returns false.
not-gt	Greater than returns false.

Example

```
<if-op-property op="available" name="myLocalVariable"/>
<if-op-property op="equal" mode="nocase" name="myProperty">true</if-local-
variable>
```

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
mode	case | nocase | regex | src-dn | dest-dn | numeric | octet Comparison mode if op="equal" or op="not-equal".	nocase
name	CDATA Name of the operation property. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false
op	available | equal | lt | gt | not-available | not-equal | not-lt | not-gt Test operator.	#REQUIRED

Content Rule

(#PCDATA)

Parent Elements

Element	Description
and	Logical conjunction.
or	Logical disjunction.

if-operation

Performs a test on the name of the current operation. The type of test performed depends on the operator specified by the op attribute. The following table shows the type of test performed by each operator.

Remarks

Operator	Returns true when...
equal	The name of the current operation is exactly equal to content of <if-operation>. Supports variable expansion.
lt	The name of the current operation is less than content of the condition when compared using the specified comparison mode. Supports variable expansion.
gt	The name of the current operation is greater than content of the condition when compared using the specified comparison mode. Supports variable expansion.
not-equal	Equal returns false.
not-lt	Less than returns false.
not-gt	Greater than returns false.

Example

```
<if-operation op="equal">add</if-operation>
```

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
mode	case| nocase | regex | scr-dn | dest-dn | numeric | octet Comparison mode if op implies a comparison.	case
notrace	true | false True if this element should not be traced during execution of the policy.	false
op	equal | lt | gt | not-equal | not-lt | not-gt Test operator.	#REQUIRED

Content Rule

(#PCDATA)

Parent Elements

Element	Description
and	Logical conjunction.
or	Logical disjunction.

if-password

Performs a test on a password in the current operation. The type of test performed depends on the operator specified by the op attribute. The following table shows the type of test performed by each operator.

Remarks

Operator	Returns true when...
available	There is password available in the current operation.
equal	There is password available in the current operation and its value equals the content of the condition when compared using the specified comparison mode. Supports variable expansion.
lt	There is password available in the current operation and its value is less than the content of the condition when compared using the specified comparison mode. Supports variable expansion.
gt	There is password available in the current operation and its value is greater than the content of the condition when compared using the specified comparison mode. Supports variable expansion.
not-available	Available returns false.
not-equal	Equal returns false.
not-lt	Less than returns false.
not-gt	Greater than returns false.

Example

```
<if-password op="available"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
mode	case| nocase | regex | scr-dn | dest-dn | numeric | octet Comparison mode if op implies a comparison.	case
notrace	true | false True if this element should not be traced during execution of the policy.	false
op	available | equal | lt | gt | not-available | not-equal | not-lt | not-gt Test operator.	#REQUIRED

Content Rule

Empty

Parent Elements

Element	Description
and	Logical conjunction.
or	Logical disjunction.

if-src-attr

Performs a test on attribute values of the current object in the source data store. The type of test performed depends on the operator specified by the op attribute. The following table shows the type of test performed by each operator.

Remarks

Operator	Returns true when...
available	There is a value available in the source data store for the specified attribute.
equal	There is a value available in the source data store for the specified attribute that equals the content of <if-src-attr> when compared using the specified comparison mode. If mode="structured", then the content must be a set of <component> elements; otherwise, it must be text. Supports variable expansion.
lt	There is a value available in the source data store for the specified attribute that is less than the content of the condition when compared using the specified comparison mode. If mode="structured" then the content must be a set of <component> elements, otherwise it must be text. Supports variable expansion.
gt	There is a value available in the source data store for the specified attribute that is greater than the content of the condition when compared using the specified comparison mode. If mode="structured" then the content must be a set of <component> elements, otherwise it must be text. Supports variable expansion.
not-available	Available returns false.
not-equal	Equal returns false.
not-lt	Less than returns false.
not-gt	Greater than returns false.

Example

```
<if-src-attr op="available" name="OU"/>
<if-src-attr op="equal" mode="nocase" name="OU">Sales</if-src-attr>
<if-src-attr op="equal" mode="structured" name="Language">
 <component name="string">EN</component>
 <component name="string">JP</component>
</if-src-attr>
```

Allowed Content

#PCDATA

Element	Description
component	Value component.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
mode	case | nocase | regex | src-dn | dest-dn | numeric | octet | structured Comparison mode if op implies a comparison.	nocase
name	CDATA Name of the attribute. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false
op	available | equal | lt | gt | not-available | not-equal | not-lt | not-gt Test operator.	#REQUIRED

Content Rule

(#PCDATA | component) *

Parent Elements

Element	Description
and	Logical conjunction.
or	Logical disjunction.

if-src-dn

Performs a test on the source DN in the current operation.

Remarks

Operator	Returns true when...
available	There is a source DN available.
equal	There is a source DN available and it equals the content of <if-src-dn> when compared using semantics appropriate to the DN format of the source data store. Supports variable expansion.
in-container	There is a source DN available and it represents an object in the container specified by the content of <if-src-dn> when compared using semantics appropriate to the DN format of the source data store.
in-subtree	There is a source DN available and it represents an object in the subtree specified by the content of <if-src-dn> when compared using semantics appropriate to the DN format of the source data store.
not-available	Available returns false.
not-equal	Equal returns false.
not-lt	Less than returns false.
not-gt	Greater than returns false.

Example

```
<if-src-dn op="available"/>
<if-src-dn op="equal">Novell\Users\Fred</if-src-dn>
<if-src-dn op="in-container">Novell\Users</if-src-dn>
<if-src-dn op="in-subtree">Novell</if-src-dn>
```

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false
op	available | equal | in-container | in-subtree | not-available | not-equal | not-in-container | not-in-subtree est operator.	#REQUIRED

Content Rule

(#PCDATA)

Parent Elements

Element	Description
and	Logical conjunction.
or	Logical disjunction.

if-xml-attr

Performs a test on an XML attribute of the current operation. The type of test performed depends on the operator specified by the op attribute. The following table shows the type of test performed by each operator.

Remarks

Operator	Returns true when...
available	There is an XML attribute with the specified name on the current operation.
equal	There is a an XML attribute with the specified name on the current operation and its value equals the content of the condition when compared using the specified comparison mode. Supports variable expansion.
lt	There is a an XML attribute with the specified name on the current operation and its value is less than the content of the condition when compared using the specified comparison mode. Supports variable expansion.
gt	There is a an XML attribute with the specified name on the current operation and its value is greater than the content of the condition when compared using the specified comparison mode. Supports variable expansion.
not-available	Available returns false.
not-equal	Equal returns false.
not-lt	Less than returns false.
not-gt	Greater than returns false.

Example

```
<if-xml-attr op="available" name="from-merge"/>
<if-xml-attr op="equal" mode="nocase" name="level">error</if-xml-attr>
```

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false

Attribute	Possible Values	Default Value
mode	case | nocase | regex | src-dn | dest-dn | numeric | octet Comparison mode if op implies a comparison.	nocase
name	CDATA Tag name of the XML attribute. Supports variable expansion. After expansion, must be a legal XML QName. It can contain a namespace prefix if and only if that prefix has been defined on the <policy>.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false
op	available | equal | lt | gt | not-available | not-equal | not-lt | not-gt Test operator.	#REQUIRED

Content Rule

(#PCDATA)

Parent Elements

Element	Description
and	Logical conjunction.
or	Logical disjunction.

if-xpath

Performs a test on the results of evaluating an XPath 1.0 expression. The type of test performed depends on the operator specified by the op attribute. The following table shows the type of test performed by each operator.

Remarks

Operator	Returns true when...
true	The XPath expression evaluates to true.
not-true	True returns false.

Example

```
<if-xpath op="true">add-attr[@attr-name='OU]/value[string(.) =
"Sales"]</if-xpath>
```

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
disabled	true | false	false
	True if this element is disabled.	
notrace	true | false	false
	True if this element should not be traced during execution of the policy.	
op	true | not-true	#REQUIRED
	Test operator.	

Content Rule

(#PCDATA)

Parent Elements

Element	Description
and	Logical conjunction.
or	Logical disjunction.

include

Includes the rules from the policy referenced by the name attribute at runtime into the including policy as if they are part of the including policy at the point of inclusion.

Remarks

The name attribute should be the slash form DN of the object containing the policy to be included. The DN might be relative to the including policy.

The inclusion is recursive because a policy might include other policies. It is an error for a policy to directly or indirectly include itself.

Example

See <policy>.

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
disabled	true | false	false
	True if this element is disabled.	
name	CDATA	#REQUIRED
	The name of the policy to include.	

Content Declaration

Empty

Parent Elements

Element	Description
policy	A policy.

or

Specifies a set of tests that are to be performed and whose results are to be logically ORed together. A set of <or> elements enclosed by a <conditions> are ANDed together.

Example

See <policy>.

Allowed Content

Element	Description
if-association	Tests an association.
if-attr	Tests an attribute in the current operation or the current object in the source data store.
if-class-name	Tests the object class of the current operation.
if-dest-attr	Tests an attribute of the current object in the destination data store.
if-dest-dn	Tests the destination DN of the current operation.
if-entitlement	Tests an entitlement of the current object.
if-global-variable	Tests a global variable.
if-local-variable	Tests a local variable.
if-named-password	Tests a named password.
if-op-attr	Tests an attribute in the current operation.
if-op-property	Tests an operation property.
if-operation	Tests the name of the current operation.
if-password	Tests the password of the current operation.
if-src-attr	Tests an attribute of current object in the source data store.
if-src-dn	Tests the source DN of the current operation.
if-xml-attr	Tests an XML attribute of the current operation.
if-xpath	Tests an XPath expression.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(if-association | if-attr | if-class-name | if-dest-attr | if-dest-dn | if-entitlement | if-global-variable | if-local-variable | if-named-password | if-op-attr | if-op-property | if-operation | if-password | if-src-attr | if-src-dn | if-xml-attr | if-xpath) *

Parent Elements

Element	Description
arg-conditions	Conditions argument.
conditions	Conditions under which the actions of a <rule> are performed.

policy

A policy consists of an ordered set of <rule>s. A <rule> consists of a set of <conditions> to be tested and an ordered set of <actions> to be performed when the conditions are met.

Remarks

A <policy> operates on an XDS document and its primary purpose is to examine and modify that document. A <policy> can also get additional context from outside of the document and cause side effects that are not reflected in the result document.

The following outline describes the basic operation of a <policy>:

- The XDS document is divided into its constituent operations. An operation is any element that is a child of <input> or <output>. An operation usually represents an event, a command, or a status.
- The <policy> is applied to separately to each operation. As the <policy> is applied to each operation in turn, that operation becomes the current operation. The object that is described by the current operation src-dn, src-entry-id, dest-dn, dest-entry-id, and/or association from the becomes the current object.
- Each <rule> is applied in order to the current operation. All of the <rule>s are applied to the current operation unless an action is performed by a prior <rule> that causes subsequent <rule>s to no longer be applied.
- The <conditions> for the <rule> are tested and if the <conditions> for the <rule> are met then the <actions> are applied.

Variables

DirXML Script supports two kinds of variables: global and local. A global variable is a variable that gets there values from a Global Configuration Value that is defined for the driver or the driver set. Global variables are by definition read-only. A local variable is a variable that is set by a policy. A local variable can exist in one of two different scopes: policy or driver. A policy scoped variable is only visible during the processing of the current operation by the policy that sets the variable. A driver scoped variable is visible from all DirXML Script policies running within the same driver until the driver is stopped. A variable name must be a legal XML Name.

There are a number of global and local variables that are automatically defined:

Name	Type	Description
dirxml.auto.driverdn	global/string	Slash format DN of the current driver
dirxml.auto.driverguid	global/string	GUID of the current driver.
dirxml.auto.treename	global/string	Tree name of the local eDirectory instance.
fromNds	policy local/boolean	True if the source data store is eDirectory. False if the source data store is the connected application.

Name	Type	Description
destQueryProcessor	policy local/java object	Instance of XdsQueryProcessor used to query the destination data store.
srcQueryProcessor	policy local/java object	Instance of XdsQueryProcessor used to query the destination data store.
destCommandProcessor	policy local/java object	Instance of XdsCommandProcessor used to query the destination data store.
srcCommandProcessor	policy local/java object	Instance of XdsCommandProcessor used to query the destination data store.
dnConverter	policy local/java object	Instance of DNConverter
current-node	policy local/node-set	The loop variable for each iteration of <do-for-each>.
current-value	policy local/node-set	The loop variable for each iteration of <do-reformat-op-attr>.
current-op	policy local/node-set	The current operation. Setting this variable using <do-set-local-variable> causes the first operation specified by <arg-node-set> to become the current operation for the remainder of the current policy execution or until it is set to another value. The new current operation must be an element sibling of the original current operation and must have been added by the current policy.

Variable Expansion

Many conditions, actions, and tokens support dynamic variable expansion in their attributes or content. Where supported, an embedded reference of the form $<variable-name>$ is replaced with the value of the local or global variable with the given name. $<variable-name>$ must be a legal variable name. If the given variable does not exist the reference is replaced with the empty string. Where it is desirable to use a single '$' and not have it interpreted as a variable reference, it should be escaped with and additional '$' (e.g. You owe me $$100.00). Content and attributes that support variable expansion are annotated with the phrase supports variable expansion.

Date/Time Parameters

Tokens that deal with dates and times have arguments that deal with the format, language, and time zone of the date and time representation. Date formats arguments may be specified in one of two ways. If the format begins with a '!' character, then the format is a named format. Legal names are defined in the following table:

Name	Description
!CTIME	Number of seconds since Midnight, January 1, 1970. (Compatible with eDirectory time syntaxes.)
!JTIME	Number of milliseconds since Midnight, January 1, 1970. (Compatible with Java time.)
!FILETIME	Number of 100-nanosecond intervals since January 1, 1601. (Compatible with Win32 FILETIME.)
!FULL.TIME	Language-specific FULL time format.
!LONG.TIME	Language-specific LONG time format.
!MEDIUM.TIME	Language-specific MEDIUM time format.
!SHORT.TIME	Language-specific SHORT time format.
!FULL.DATE	Language-specific FULL date format.
!LONG.DATE	Language-specific LONG date format.
!MEDIUM.DATE	Language-specific MEDIUM date format.
!SHORT.DATE	Language-specific SHORT date format.
!FULL.DATETIME	Language-specific FULL date/time format.
!LONG.DATETIME	Language-specific LONG date/time format.
!MEDIUM.DATETIME	Language-specific MEDIUM date/time format.
!SHORT.DATETIME	Language-specific SHORT date/time format.

If the format does not begin with '!', then the format is interpreted as a custom date/time format conforming to the patterns recognized by java.text.SimpleDateFormat.

Language arguments can be specified by an identifier that conforms to IETF RFC 3066. The list of identifiers understood by the system can be obtained by calling java.util.Locale.getAvailableLocales() and substituting all underscores in the result with a hyphens. If a language argument is omitted or blank, then the default system language is used.

Time zone arguments can be specified in any identifier recognizable by java.util.TimeZone.getTimeZone(). A list of identifies understood by the system can be obtained by calling java.util.TimeZone.getAvailableIDs(). If a time zone argument is omitted or blank, then the default system time zone is used.

XPath Evaluation

Arguments to some conditions and actions take an XPath 1.0 expression. This XPath is evaluated with the following context:

- The context node is the current operation unless otherwise specified in the description of the expression.
- The context position and size are 1.

- Available variables
  - Those available as parameters to style sheets within the Identity Manager Metadirectory engine (currently fromNds, srcQueryProcessor, destQueryProcessor, srcCommandProcessor, destCommandProcessor, and dnConverter.)
  - Global configuration variables.
  - Local policy variables.
  - If there is a name conflict between the different variable sources then the order of precedence is local (policy scope), local (driver scope), global.
  - Because of the XPath syntax, any variable that has a colon character in its name is not accessible from XPath.
- Namespaces that are declared on <policy>.
- Available functions
  - All built-in XPath 1.0 functions
  - Java extension functions as provided by NXSL.
    - Namespaces declarations to associate a prefix with a Java class must be declared on <policy>.

Example

```
<policy>
 <description>My policy</description>
 <include name="..\..\Library\My shared policy"/>
 <rule>
  <description>Rule to disallow moving a user</description>
  <comment>This rule was added because under no circumsances do we ever want to
perform a move.</comment>
  <conditions>
   <and>
    <if-class-name op="equal" mode="nocase">User</if-class-name>
    <if-operation op="equal">move</if-operation>
   </and>
  </conditions>
  <actions>
   <veto/>
  </actions>
 </rule>
 <rule>
  <description>Rule to disallow operations on a disabled user or group</
description>
  <conditions>
   <or>
    <if-class-name op="equal" mode="nocase">User</if-class-name>
    <if-class-name op="equal" mode="nocase">Group</if-class-name>
   </or>
   <or>
    <if-attr op="equal" mode="nocase" name="Login Disabled">true</if-attr>
   </or>
  </conditions>
  <actions>
   <veto/>
  </actions>
 </rule>
</policy>
```

Allowed Content

Element	Description
description	Description of a <policy> or a <rule>.
rule	Rule within a policy.
include	Include rules from another policy.

Attributes

None

Content Rule

(description ? , (rule | include) *)

Parent Elements

None

rule

Specifies a set of <actions> and a set of <conditions> under which those <actions> are performed.

Example

See <policy>.

Allowed Content

Element	Description
description	Description of a <policy> or a <rule>.
comment	Long description of a <rule>.
conditions	Conditions under which the actions of a <rule> are performed.
actions	Actions that are performed by a <rule>.

Attributes

Attribute	Possible Values	Default Value
disabled	true | false True if this element is disabled.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(description ? , comment * , conditions , actions)

Parent Elements

Element	Description
policy	A policy.

token-added-entitlement

Expands to the granted values of the named entitlement in the current operation. If its parent element is <arg-node-set>, then all the available values are returned as <entitlement-impl> elements in a node set. Otherwise, the first available value is returned as a string.

Example

```
<token-added-entitlement name="manager"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
name	CDATA Name of the variable. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.

Element	Description
token-convert-time	Converts a date/time from one format to another format.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-join	Joins a node set into a string.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node-set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-association

Expands to the association value specified in the current operation.

Example

```
<token-association/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-convert-time	Converts a date/time from one format to another.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-join	Joins a node set into a string.

Element	Description
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-attr

Expands to the attribute values of the current object in either the current operation or the source data store. It can logically be thought of as the union of <token-op-attr> and <token-src-attr>. If its parent element is <arg-node-set> then all the available <value> elements are returned as nodes in a node set. Otherwise the first available value is returned as a string.

Example

```
<token-attr name="OU"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
name	CDATA Name of the attribute. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decodes base64 data into a string.

Element	Description
token-base64-encode	Encodes a string into base64 data.
token-convert-time	Converts a date/time from one format to another.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-join	Joins a node set into a string.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-base64-decode

Decodes the result of the enclosed tokens from base64 encoded data to bytes and then converts the bytes into a string using the character set specified by the character set.

Example

```
<token-base64-decode charset="UTF-8">
 <token-op-attr name="data"/>
</token-base64-decode>
```

Allowed Content

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-convert-time	Convert a date/time from one format to another.
token-escape-for-dest-dn	Converts a string for use in a source destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-join	Joins a node set into a string.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-parse-dn	Parses and converts a DN.
token-replace-all	Replaces all instances of substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

Attributes

Attribute	Possible Values	Default Value
charset	CDATA The character set used to convert the decoded bytes to a string. If not specified, the encoding specified by the system property file.encoding is used. Supports variable expansion.	#IMPLIED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-generate-password | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) +

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument
arg-value	Value argument.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-convert-time	Converts a date/time from one format to another.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.

Element	Description
token-join	Joins a node-set into a string.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-base64-encode

Converts the result of the enclosed tokens to bytes using the character set specified by the character set, and then base64 encodes the bytes.

Example

```
<token-base64-encode charset="UTF-8">
 <token-op-attr name="Surname"/>
</token-base64-encode>
```

Allowed Content

Elements	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or the current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name for the current operation.
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of the current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lower case.
token-map	Maps a string through a mapping table.

Elements	Description
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	A generated unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

Attribute	Possible Values	Default Value
charset	CDATA The character set used to convert the string to bytes. If not specified, the encoding specified by the system property file.encoding is used. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-generate-password | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) +

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-convert-time	Converts a date/time from one format to another.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.

Element	Description
token-join	Joins a node set into a string.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-char

Expands to the character specified by the Unicode code point specified by the value.

Example

```
<token-char value="10"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false
value	CDATA The Unicode code point of the character. Supports variable expansion.	#REQUIRED

Content Declaration

Empty

Parent Elements

Element	Description
arg-actions	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-convert-time	Converts a date/time from one format to another.

Element	Description
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-join	Joins a node set into a string.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-class-name

Expands to the object class name specified in the current operation.

Example

```
<token-class-name/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument
arg-value	Value argument.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-convert-time	Converts a date/time from one format to another.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-join	Joins a node-set into a string.

Element	Description
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-convert-time

Converts the date/time represented by the result of the enclosed tokens from the format, language and time zone specified by src-format, src-lang, and src-tz, to the format, language, and time zone specified by dest-format, dest-lang, and dest-tz and adds an optional offset time specified by offset and offset-unit. See "Date/Time Parameters" on page 361 for information on specifying formats, languages, and time zones.

Example

```
<token-convert-time src-format="MM/dd/YYYY" src-lang="en-US" src-tz="MST" dest-
format="dd/MM/YYYY" src-lang="en-US" src-tz="MST">
 <token-op-attr name="birthdate"/>
</token-convert-time>
```

Allowed Content

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.

Element	Description
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

Attribute	Possible Values	Default Value
dest-format	CDATA The destination date/time format. Supports variable expansion.	#REQUIRED
dest-lang	CDATA The destination language (defaults to the current system language). Supports variable expansion.	#IMPLIED
dest-tz	CDATA The destination time zone (defaults to the current system time zone). Supports variable expansion.	#IMPLIED
notrace	true | false True if this element should not be traced during execution of the policy.	false
offset	CDATA The time offset (a negative number subtracts the time interval). Supports variable expansion.	#IMPLIED
offset-unit	second | minute | hour | day | week | month | year The units of the time offset. Supports variable expansion.	day
src-format	CDATA The source date/time format. Supports variable expansion.	#REQUIRED
src-lang	CDATA The source language (defaults to the current system language). Supports variable expansion.	#IMPLIED
src-tz	CDATA The source time zone (defaults to the current system time zone). Supports variable expansion.	#IMPLIED

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-generate-password | token-global-variable | token-join | token-local-variable |

token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath)

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument
arg-value	Value argument.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-convert-time	Converts a date/time from one format to another.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-join	Joins a node-set into a string.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-dest-attr

Expands to the attribute values of the current object in the destination data store. If its parent element is <arg-node-set>, then all the available <value> elements are returned as nodes in a node set. Otherwise, the first available value is returned as a string.

Example

```
<token-dest-attr name="OU"/>
```

Allowed Content

Element	Description
arg-dn	DN argument.
arg-association	Association argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA Class name of the object to read. Can be required (for schema mapping purposes) if the object is other than the current object.	#IMPLIED
name	CDATA Name of the attribute.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(arg-dn | arg-association) ?

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.

Element	Description
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument
arg-value	Value argument.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-convert-time	Converts a date/time from one format to another.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-join	Joins a node-set into a string.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-dest-dn

Expands to the destination DN specified in the current operation or a portion thereof. If start and length are not specified or are set to the default values {0,-1}, then the entire DN is used; otherwise only the portion of the DN specified by start and length is used. The format of the DN is automatically set to the format of the source data store if convert attribute is set to true.

Example

```
<token-dest-dn/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
convert	true | false Converts to the DN format of source data store.	false
length	CDATA The number of DN segments to include negative numbers are interpreted as (total # of segments + length) + 1. For example, for a DN with 5 segments, a length of -1 = (5 + (-1)) + 1 = 5, -2 = (5 + (-2)) + 1 = 4, etc.	-1
notrace	true | false True if this element should not be traced during execution of the policy.	false
start	CDATA The segment index to start with 0 is the rootmost segment. >0 is an offset from the rootmost segment. -1 is the leafmost segment. <-1 is an offset from the leafmost segment towards the rootmost segment.	0

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument
arg-value	Value argument.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-convert-time	Converts a date/time from one format to another.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-join	Joins a node-set into a string.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-dest-name

Expands to the unqualified RDN of the destination DN specified in the current operation.

Example

```
<token-dest-name/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(#PCDATA)

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument
arg-value	Value argument.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-convert-time	Converts a date/time from one format to another.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-join	Joins a node-set into a string.

Element	Description
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-document

Reads the XML document pointed to by the URI specified by the arg-string and returns the document node in a node set. The URI can be relative to the URI of the including policy. If the URI or DN cannot be resolved to a well-formed XML document, the result is an empty node set.

Example

```
<token-document>
   <arg-string>
      <token-text>../MyDriver#DirXML-DriverFilter</token-text>
   </arg-string>
</token-document>
```

Allowed Content

Element	Description
arg-string	String argument.

Attributes

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(arg-string)

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument
arg-value	Value argument.

Element	Description
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-convert-time	Converts a date/time from one format to another.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-join	Joins a node-set into a string.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-entitlement

Expands to the granted values of the named entitlement for the current object. If its parent element is <arg-node-set>, then all the available values are returned as <entitlement-impl> elements in a node set. Otherwise, the first available value is returned as a string.

Example

```
<token-entitlement name="manager"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
name	CDATA Name of the attribute. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument
arg-value	Value argument.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.

Element	Description
token-convert-time	Converts a date/time from one format to another.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-join	Joins a node-set into a string.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-escape-for-dest-dn

Expands to a version of the expansion of the concatenation of the enclosed tokens, which has been escaped for use in a DN according to the rules of the destination DN format.

Example

```
<token-escape-for-dest-dn>
 <token-attr name="Surname"/>
</token-escape-for-dest-dn>
```

Allowed Content

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.

Element	Description
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-generate-password | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) +

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument
arg-value	Value argument.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-convert-time	Converts a date/time from one format to another.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-join	Joins a node-set into a string.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-parse-dn	Parses or converts a DN.

Element	Description
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-escape-for-src-dn

Expands to a version of the expansion of the concatenation of the enclosed tokens that has been escaped for use in a DN according to the rules of the source DN format.

Example

```
<token-escape-for-src-dn>
 <token-attr name="Surname"/>
</token-escape-for-src-dn>
```

Allowed Content

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.

Element	Description
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-generate-password | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) +

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument
arg-value	Value argument.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-convert-time	Converts a date/time from one format to another.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-join	Joins a node-set into a string.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-parse-dn	Parses or converts a DN.

Element	Description
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-generate-password

If a password policy is specified for this token, NMAS uses this policy for generating passwords. For successful password generation, this password policy must exclude the rules that requires a user object lookup in eDirectory. For example, you should not use a policy to generate passwords using the generate password token if it requires the newly created password to be different from the user's password history that needs eDirectory user object lookup for comparison. Instead, create a new password policy similar to the existing password policy and exclude the rules that require a user lookup in eDirectory. Do not assign the new password policy to the user container. Instead, use it only to generate the random password for users from the Identity Manager policies when the user is added.

Example

```
<token-generate-password policy-dn="..\my password policy"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false
policy-dn	CDATA Slash form DN of a nspmPasswordPolicy object. Can be relative to the including policy. Supports variable expansion.	#IMPLIED

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.

Element	Description
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-global-variable

Expands to the value of the named global configuration variable.

Example

```
<token-global-variable name="Fred"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
name	CDATA Name of the variable. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.

Element	Description
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-join

Joins the values of the nodes in the node-set result of the enclosed tokens, separating the values by the characters specified by the delimiter. If csv is true, then CSV quoting rules are applied to the values.

Example

```
<!-- combine all of the members of the group into a CSV record -->
<token-join delimiter="," csv="true">
 <token-op-attr name="Member"/>
</token-join>
```

Allowed Content

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.

Element	Description
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

Attribute	Possible Values	Default Value
csv	true | false Applies CSV quoting to values.	false
delimiter	CDATA The string use to delimit the joined values. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-generate-password | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath | token-query | token-split) +

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.

Element	Description
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-local-variable

Expands to the value of the named local variable. If its parent element is not <arg-node-set> and the variable holds a node set, then the string value of the node set is returned. If the same local variable exists in both the policy scope and the driver scope, the variable in the policy scope takes precedence.

Example

```
<token-local-variable name="myVariable"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
name	CDATA Name of the variable. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.

Element	Description
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-lower-case

Converts each character of the result of the enclosed tokens to lowercase.

Example

```
<token-lower-case>
 <token-attr name="Surname"/>
</token-lower-case>
```

Allowed Content

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.

Element	Description
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

Attribute	Possible Values	Default Value
name	CDATA Name of the attribute. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-generate-password | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) +

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.

Element	Description
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-map

Maps the result of the enclosed tokens from the values specified by the src column to the dest column in the mapping table specified by table.

Remarks

The table attribute should be the slash form DN of the DirXML-Resource object containing the mapping table to be used. The DN may be relative to the including policy.

If this token is evaluated in a context where a node set result is expected and multiple rows are matched by the value being mapped, then a node set is returned that contains the values from the destination column of each matching row. Otherwise only the value from the first matching row is returned.

If no rows are matched by the value being mapped and a non-empty value for default-value is provided, then the token returns the value of default-value, otherwise it returns the empty string if being evaluated in a context that is expecting a string, or an empty node-set if evaluated in a context that is expecting a node-set.

Example

```
<token-map table="./Department Table" src="dept" dest="code">
 <token-op-attr name="OU"/>
</token-map>
```

Allowed Content

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.

Element	Description
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.

Element	Description
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

Attribute	Possible Values	Default Value
default-value	CDATA Default value for the destination column. Supports variable expansion.	#IMPLIED
dest	CDATA Name of the destination column. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false
src	CDATA Name of the source column. Supports variable expansion.	#REQUIRED
table	CDATA Slash form DN of a DirXML-Resource object containing the mapping table. Can be relative to the including policy. Supports variable expansion.	#REQUIRED

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-generate-password | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) +

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-named-password

Expands to the named password from the driver.

Example

```
<token-named-password name="extraPassword"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
name	CDATA Name of the attribute. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.

Element	Description
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-op-attr

Expands to the attribute values in the current operation (<add-attr>, <add-value> or <attr>). If it's parent element is <arg-node-set>, then all the available <value> elements are returned as nodes in a node set. Otherwise, the first available value is returned as a string.

Example

```
<token-op-attr name="OU"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
name	CDATA Name of the attribute. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.

Element	Description
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-op-property

Expands to the value of the named operation property on the current operation.

Example

```
<token-op-property name="myProperty"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
name	CDATA Name of the attribute. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.

Element	Description
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-operation

Expands to the name of the current operation.

Example

```
<token-operation/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.

Element	Description
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-parse-dn

Expands to a version of the DN specified by expansion of the concatenation of the enclosed tokens. The DN is parsed according the format specified by <src-dn-format>. The portion of the DN specified by start and length is then converted to the format specified by <dest-dn-format>.

Remarks

<src-dn-delims> and <dest-dn-delims> are used to specify custom DN formats. The eight characters that make up the delimiter set are defined as follows:

- Typed Name Boolean Flag: 0 means names are not typed, 1 means names are typed
- Unicode No-Map Character Boolean Flag: 0 means don't output or interpret unmappable Unicode characters as escaped hex digit strings, for example \FEFF. The Unicode characters 0xfeff, 0xfffe, 0xfffd, and 0xffff are not accepted by eDirectory.
- Relative RDN Delimiter
- RDN Delimiter
- Name Divider
- Name Value Delimiter
- Wildcard Character
- Escape Character

If RDN Delimiter and Relative RDN Delimiter are the same character, then the orientation of the name is root right, otherwise the orientation is root left.

If there are more than eight characters in the delimiter set, the extra characters are all considered to be characters that need to be escaped but have no other special meaning.

Example

```
<token-parse-dn src-dn-format="src-dn" dest-dn-format="dest-dn" start="0"
length="-1">
   <token-op-attr name="Group Membership"/>
</token-parse-dn>
```

Allowed Content

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation

Element	Description
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.

Element	Description
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

Attribute	Possible Values	Default Value
dest-dn-delims	CDATA Specifies the custom DN delimiter set when dest-dn-format="custom".	#IMPLIED
dest-dn-format	src-dn | dest-dn | dot | qualified-dot | slash | qualified-slash | ldap | custom The format used to output the parsed DN.	dest-dn
length	CDATA The number of DN segments to include negative numbers are interpreted as (total # of segments + length) + 1. For example, for a DN with 5 segments a length of -1 = (5 + (-1)) + 1 = 5, -2 = (5 + (-2)) + 1 = 4, etc.	-1
notrace	true | false True if this element should not be traced during execution of the policy.	false
src-dn-delims	CDATA Specifies the custom DN delimiter set when src-dn-format="custom"	#IMPLIED

Attribute	Possible Values	Default Value
src-dn-format	src-dn | dest-dn | dot | qualified-dot | slash | qualified-slash | ldap | custom The format used to parse the enclosed DN.	src-dn
start	CDATA The segment index to start with: 0 is the rootmost segment. >0 is an offset from the rootmost segment. -1 is the leafmost segment. <-1 is an offset from the leafmost segment towards the rootmost segment.	0

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-generate-password | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) +

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.

Element	Description
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-password

Expands to the password specified in the current operation.

Example

```
<token-password/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.

Element	Description
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-query

Causes a <query> to be performed in the source or destination data store and returns the resulting <instance> elements.

Remarks

The data store to search is specified by the data store.

The base of the query is specified by either <arg-dn> or <arg-association>. If neither are specified, then the base is the root of the data store.

The scope of the query is specified by query.

The class of the query is specified by class-name. If omitted the query looks for all classes.

The set of attributes to search for is specified by the <arg-match-attr> elements.

The set of attributes to return is specified by the <arg-string> elements. If no <arg-string> elements are specified then no attributes are read. If one of the <arg-string> evaluates to the asterisk character, then all attributes are read.

If max-result-count is specified, then <query-ex> is issued instead of a <query> and the results are returned in batches. When used in the context of a <do-for-each> subsequent batches (if any) are automatically retrieved.

Example

```
<token-query scope="subordinates" class-name="User" datastore="dest">
   <arg-dn>
      <token-text>Users/</token-text>
      <token-attr name="OU"/>
   </arg-dn>
   <arg-match-attr name="CN"/>
   <arg-match-attr name="L"/>
      <arg-value>
         <token-text>Provo</token-text>
      </arg-value>
   </arg-match-attr>
   <arg-string>
      <token-text>Surname</token-text>
   </arg-string>
   <arg-string>
     <token-text>Given Name</token-text>
   </arg-string>
</token-query>
```

Allowed Content

Element	Description
arg-association	Association argument.
arg-dn	DN argument.
arg-match-attr	Matches the attribute argument.
arg-string	String argument

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA The class name of the query. Supports variable expansion.	#IMPLIED
datastore	src | dest The data store to be queried.	dest
max-result-count	CDATA The maximum number of results to return per batch.	#IMPLIED
notrace	true | false True if this element should not be traced during execution of the policy.	false
scope	entry | subordinates | subtree The scope of the query.	subtree

Content Rule

((arg-dn | arg-association) ? , arg-match-attr * , arg-string *)

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.

Element	Description
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-removed-attr

Expands to the attribute values removed in the current operation (<remove-attr>). If its parent element is <arg-node-set>, then all the available <value> elements are returned as nodes in a node-set. Otherwise, the first available value is returned as a string.

Example

```
<token-removed-attr name="OU"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
name	CDATA Name of the attribute.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.

Element	Description
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-removed-entitlement

Expands to the revoked values of the named entitlement in the current operation. If its parent element is <arg-node-set>, then all the available values are returned as <entitlement-impl> elements in a node set. Otherwise, the first available value is returned as a string.

Example

```
<token-removed-entitlement name="manager"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
name	CDATA Name of the attribute.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.

Element	Description
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-replace-all

Expands to a version of the expansion of the concatenation of the enclosed tokens where all matching instances of the regular expression specified by regex are replaced by the string specified by replace-with.

Remarks

See http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Pattern.html and http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Matcher.html#replaceAll(java.lang.String)

The pattern options CASE_INSENSITIVE, DOTALL, and UNICODE_CASE are used but can be reversed using the appropriate embedded escapes.

Example

```
<!-- remove escaping from DN in slash format -->
<token-replace-all regex="'(.)" replace-with="$1">
   <token-dest-dn/>
</token-replace-all>
```

Allowed Content

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.

Element	Description
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.

Element	Description
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false
regex	CDATA Regular expression that matches the substring to replace.	#REQUIRED
replace-with	CDATA Regular expression that specifies the replacement string.	#REQUIRED

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-generate-password | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) +

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.

Element	Description
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-replace-first

Expands to a version of the expansion of the concatenation of the enclosed tokens where the first matching instance of the regular expression specified by regex is replaced by the string specified by replace-with.

Remarks

See http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Pattern.html and http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Matcher.html#replaceAll(java.lang.String)

The pattern options CASE_INSENSITIVE, DOTALL, and UNICODE_CASE are used but can be reversed using the appropriate embedded escapes.

Example

```
<!-- change Full Name for "Surname, Given-Names" to "Given-Names Surname" -->
<token-replace-first regex="^(.*), (.*)$" replace-with="$2 $1">
   <token-attr name="Full Name"/>
</token-replace-first>
```

Allowed Content

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.

Element	Description
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.

Element	Description
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false
regex	CDATA Regular expression that matches the substring to replace.	#REQUIRED
replace-with	CDATA Regular expression that specifies the replacement string.	#REQUIRED

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-generate-password | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) +

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.

Element	Description
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-resolve

<token-split> resolved the DN specified by <arg-dn> to an association key, or the association key specified by <arg-association> to a DN in the data store specified by data store.

Example

```
<token-resolve datastore="src">
 <arg-dn>
  <token-op-attr name="manager"/>
 </arg-dn>
</token-resolve>
```

Allowed Content

Element	Description
arg-association	Association argument.
arg-dn	DN argument.

Attributes

Attribute	Possible Values	Default Value
datastore	src | dest The data store to be queried.	dest
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(arg-dn | arg-association)

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.

Element	Description
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-split

Splits the result of the enclosed tokens into a node set consisting of text nodes based on the pattern specified by the delimiter. If csv is true, then CSV quoting rules will be honored during the parsing of the string.

Example

```
<token-split delimiter="," csv="true">
 <token-text>Doe,John,"Doe, John"</toke-text>
</token-join>
```

Allowed Content

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.

Element	Description
token-map	Maps a string through a mapping table.
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

Attribute	Possible Values	Default Value
csv	true | false Honor CSV style quoting.	false
delimiter	CDATA Regular expression that matches the delimiter characters. Supports variable expansion.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-generate-password | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) +

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.

Element	Description
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-src-attr

Expands to the attribute values of the current object in the source data store. If its parent element is <arg-node-set>, then all the available <value> elements are returned as nodes in a node set. Otherwise, the first available value is returned as a string.

Example

```
<token-src-attr name="OU"/>
```

Allowed Content

Element	Description
arg-association	Association argument.
arg-dn	DN argument.

Attributes

Attribute	Possible Values	Default Value
class-name	CDATA Class name of the object to read. Can be required (for schema mapping purposes) if the object is other than the current object.	#IMPLIED
name	CDATA Name of the attribute.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(arg-dn | arg-association) ?

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.

Element	Description
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-src-dn

Expands to the source DN specified in the current operation or a portion thereof. If start and length are not specified or are set to the default values {0,-1}, then the entire DN is used; otherwise, only the portion of the DN specified by start and length is used. The format of the DN is converted to the format of the destination data store if the convert attribute is set to true.

Example

```
<token-src-dn/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
convert	true | false Convert to the DN format of the destination data store.	false
length	CDATA The number of DN segments to include negative numbers are interpreted as (total # of segments + length) + 1. For example, for a DN with 5 segments a length of -1 = (5 + (-1)) + 1 = 5, -2 = (5 + (-2)) + 1 = 4, etc.	-1
notrace	true | false True if this element should not be traced during execution of the policy.	false
start	CDATA The segment index to start with: 0 is the rootmost segment. >0 is an offset from the rootmost segment. -1 is the leafmost segment. <-1 is an offset from the leafmost segment towards the rootmost segment.	0

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-src-name

Expands to the unqualified RDN of the source DN specified in the current operation.

Example

```
<token-src-name/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.

Element	Description
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-substring

Expands to a substring of the expansion of the concatenation of the enclosed tokens.

Example

```
<token-substring start="0" length="1">
 <token-attr name="Given Name"/>
</token-substring>
```

Allowed Content

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.

Element	Description
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

Attribute	Possible Values	Default Value
length	CDATA The number of characters to include negative numbers are interpreted as (total # of characters + length) + 1. For example, for a string with 5 characters a length of -1 = (5 + (-1)) + 1 = 5, -2 = (5 + (-2)) + 1 = 4, etc.	-1
notrace	true | false True if this element should not be traced during execution of the policy.	false
start	CDATA The character index to start with: 0 is the first character. >0 is an offset from the start if the string. -1 is the last character.<-1 is an offset from the last character towards the start of the string of the string.	0

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-generate-password | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) +

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.

Element	Description
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-text

Expands to the enclosed text.

Example

```
<token-text>Fred</token-text>
```

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(#PCDATA)

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.

Element	Description
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-time

Expands to the current date/time in the format, language and time zone specified by format, language, and time zone. See "Date/Time Parameters" on page 361 for information on specifying formats, languages, and time zones.

Example

```
<token-time src-format="!CTIME" tz="UTC"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
format	CDATA The date/time format. Supports variable expansion.	#REQUIRED
lang	CDATA The language (defaults to the current system language). Supports variable expansion.	#IMPLIED
notrace	true | false True if this element should not be traced during execution of the policy.	false
tz	CDATA The time zone (defaults to the current system time zone). Supports variable expansion.	#IMPLIED

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.

Element	Description
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-unique-name

Expands to a pattern based name that is unique in the destination data store according to the criteria specified.

Remarks

Each <arg-string> element provides a pattern to be used to create a proposed name.

A proposed name is tested by performing a query for that value in the name attribute against the destination data store using the <arg-dn> element or the <arg-association> element as the base of the query and scope as the scope of the query. If the destination data store is the Identity Vault and name is omitted, then a search is performed against the pseudo-attribute “[Entry].rdn”, which represents the RDN of an object without respect to what the naming attribute might be. If the destination data store is the application, then name is required.

A pattern can be tested with and/or without a counter as indicated by counter-use and counter-pattern. When a pattern is tested with a counter, the pattern is tested repeatedly with an appended counter until a name is found that does not return any instances or the counter is exhausted. The counter starting value is specified by counter-start and the counter maximum value is specified in terms of the maximum number of digits as specified by counter-digits. If the number of digits is less than those specified, then the counter is right padded with zeros unless the counter-pad attribute is set to false. The counter is considered exhausted when the counter can no longer be represented by the specified number of digits.

As soon as a proposed name is determined to be unique, the testing of names is stopped and the unique name is returned.

The order of proposed names is tested as follows:

- Each pattern is tested in the order specified. If counter-use=“always” and the pattern is one of the patterns indicated by the counter-pattern then the pattern is tested with a counter, otherwise it is tested without a counter.
- If no unique name has been found after the patterns have been exhausted and counter-use=“fallback”, then the patterns indicated by the counter-pattern are retried with a counter.

If all specified combinations of patterns and counters are exhausted, then the action specified by the on-unavailable is taken.

Example

```
<token-unique-name counter-digits="2" counter-pad="true" counter-pattern="first"
counter-start="1" counter-use="fallback" name="CN" on-unavailable="error"
scope="subtree" test-all-objects="true">
  <arg-string>    <token-upper-case>
      <token-substring length="1" start="0">
        <token-attr name="Given Name"/>
      </token-substring>
      <token-attr name="Surname"/>
    </token-upper-case>
  </arg-string>
  <arg-string>
    <token-upper-case>
      <token-substring length="1" start="0">
        <token-attr name="Given Name"/>
```

```
      </token-substring>
      <token-substring length="1" start="0">
        <token-attr name="MI"/>
      </token-substring>
      <token-attr name="Surname"/>
    </token-upper-case>
  </arg-string>
  <arg-string>
    <token-upper-case>
      <token-attr name="Given Name"/>
      <token-attr name="Surname"/>
    </token-upper-case>
  </arg-string>
</token-unique-name>
```

Allowed Content

Element	Description
arg-association	Association argument.
arg-dn	DN argument.
arg-string	String argument

Attributes

Attribute	Possible Values	Default Value
counter-digits	CDATA	#IMPLIED
	Width in digits of counter.	
counter-pad	true | false	true
	Enable/disable right zero padding of counter.	
counter-pattern	first | last | all	last
	Which patterns to use counter with:	
	First: Use counter only with the first pattern.	
	Last: Use counter only with the last pattern.	
	All: Use counter with all patterns.	
counter-start	CDATA	1
	Number to start counter.	

Attribute	Possible Values	Default Value
counter-use	always | never | fallback	fallback
	When to use counters:	
	Never: Don't use counters.	
	Always: Always use counters on the patterns indicated by counter-pattern.	
	Fallback: Use counters counter the patterns indicated by counter-pattern only after all patterns have failed without counters.	
name	CDATA	#IMPLIED
	Name of attribute to check for uniqueness.	
notrace	true | false	false
	True if this element should not be traced during execution of the policy.	
on-unavailable	ignore | warning | error | fatal	error
	Action to take if unique name cannot be constructed:	
	Ignore: Ignore and return empty name.	
	Warning: Issue warning and return empty name.	
	Error: Generate error and abort current transaction.	
	Fatal: Generate fatal error and shut down driver.	
scope	subordinates | subtree	subtree
	The scope in which to check uniqueness.	
test-all-objects	true | false	false
	The include/exclude object class name in the unique-name query.	

Content Rule

((arg-dn | arg-association) ? , arg-string +)

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-unmatched-src-dn

Expands to the portion of the source DN in the current operation that corresponds to the part of the DN that was not matched by the most recent match of an <if-src-dn> in the conditions for this rule (taking into account short circuit evaluation). If there were no matches, then the entire DN is used. The format of the DN is converted to the format of the destination data store if the convert attribute is set to true.

Example

```
<token-unmatched-src-dn/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
convert	true | false Convert to the DN format of the destination data store.	false
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.

Element	Description
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-upper-case

Expands to a version of the expansion of the concatenation of the enclosed tokens with each character converted to uppercase.

Example

```
<token-upper-case>
 <token-attr name="Surname"/>
</token-upper-case>
```

Allowed Content

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.

Element	Description
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-generate-password | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) +

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.

Element	Description
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-xml-parse

Parses the result of the enclosed tokens as XML and returns the resulting document node in a node set. If the result of the enclosed tokens is not well-formed XML or cannot be parsed for any reason, an empty node set is returned.

Example

```
<token-xml-parse>
 <token-base64-decode charset="UTF-8">
  <token-op-attr name="data"/>
 </token-base64-decode>
</token-xml-parse>
```

Allowed Content

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.

Element	Description
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-generate-password | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) +

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.

Element	Description
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-xml-serialize

Serializes the node set result enclosed tokens as XML. Depending on the content of the node set, the resulting string is either a well-formed XML document or a well-formed parsed general entity. A parsed general entity cannot be parsed as a standalone XML document.

Example

```
<token-xml-serialize>
 <token-xpath expression="."/>
</token-xml-serialize>
```

Allowed Content

Element	Description
token-added-entitlement	The values of an entitlement granted in the current operation.
token-association	The association value from the current operation.
token-attr	The values of an attribute in the current operation or current object in the source data store.
token-base64-decode	Decodes base64 data into a string.
token-base64-encode	Encodes a string into base64 data.
token-char	A Unicode character.
token-class-name	The object class name from the current operation
token-convert-time	Converts a date/time from one format to another.
token-dest-attr	The values of an attribute of current object in the destination data store.
token-dest-dn	A value derived from the destination DN from the current operation.
token-dest-name	The unqualified RDN derived from the destination DN from the current operation.
token-document	Reads an XML document.
token-entitlement	The values of a granted entitlement of the current object.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-generate-password	Generates a random password.
token-global-variable	The value of a global variable.
token-join	Joins a node-set into a string.
token-local-variable	The value of a local variable.
token-lower-case	Converts a string to lowercase.

Element	Description
token-map	Maps a string through a mapping table.
token-named-password	The value of the named password.
token-op-attr	The values of an attribute in the current operation.
token-op-property	The value of an operation property.
token-operation	The name of the current operation.
token-parse-dn	Parses or converts a DN.
token-password	The value of the password in the current operation.
token-query	Queries the source or destination data store.
token-removed-attr	The values of an attribute removed in the current operation.
token-removed-entitlement	The values of an entitlement revoked in the current operation.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-resolve	Resolves a DN to an association key or an association key to a DN.
token-split	Splits a string into a node set.
token-src-attr	The values of an attribute of the current object in the source data store.
token-src-dn	A value derived from the source DN from the current operation.
token-src-name	The unqualified RDN derived from source DN from the current operation.
token-substring	Substring of a string.
token-text	Constant text.
token-time	The current date/time.
token-unique-name	Generates a unique name.
token-unmatched-src-dn	A DN relative to the one matched by if-src-dn.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.
token-xpath	The result of an XPath expression.

Attributes

Attribute	Possible Values	Default Value
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Rule

(token-added-entitlement | token-association | token-attr | token-base64-decode | token-base64-encode | token-char | token-class-name | token-convert-time | token-dest-attr | token-dest-dn | token-dest-name | token-document | token-entitlement | token-escape-for-dest-dn | token-escape-for-src-dn | token-generate-password | token-global-variable | token-join | token-local-variable | token-lower-case | token-named-password | token-map | token-op-attr | token-op-property | token-operation | token-parse-dn | token-password | token-removed-attr | token-removed-entitlement | token-replace-all | token-replace-first | token-resolve | token-src-attr | token-src-dn | token-src-name | token-substring | token-text | token-time | token-unique-name | token-unmatched-src-dn | token-upper-case | token-xml-parse | token-xml-serialize | token-xpath) +

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decode base64 data into a string.
token-base64-encode	Encode a string into base64 data.
token-convert-time	Convert a data/time from one format to another format.
token-escape-for-dest-dn	Convert a string for use in the destination DN.
token-escape-for-src-dn	Convert a string for use in a source DN.
token-join	Join a node set into a string.
token-lower-case	Convert a string to lowercase.
token-map	Map a string through a mapping table.
token-parse-dn	Parses or converts a DN.

Element	Description
token-replace-all	Replaces all instances of a substring with a string.
token-replace-first	Replaces a single instance of a substring within a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

token-xpath

Expands to results of evaluating an XPath 1.0 expression. If its parent element is <arg-node-set> and the expression returns a node set, then the node set is returned as is. If its parent element is <arg-node-set> and the expression returns a data type other than node set, then a text node is containing the string value of the result is returned. Otherwise, the string value of the result is returned.

Example

```
<token-xpath
expression="*[@attr-name='OU']//value[starts-with(string(.),'xxx')]"/>
```

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
expression	CDATA XPath expression.	#REQUIRED
notrace	true | false True if this element should not be traced during execution of the policy.	false

Content Declaration

Empty

Parent Elements

Element	Description
arg-association	Association argument.
arg-component	Component argument.
arg-dn	DN argument.
arg-node-set	Node set argument.
arg-object	Java Object argument
arg-password	Password argument.
arg-string	String argument.
arg-value	Value argument.
token-base64-decode	Decodes base64 data into a string.

Element	Description
token-base64-encode	Encodes a string into base64 data.
token-convert-time	Converts a date/time from one format to another.
token-escape-for-dest-dn	Converts a string for use in a destination DN.
token-escape-for-src-dn	Converts a string for use in a source DN.
token-join	Joins a node set into a string.
token-lower-case	Converts a string to lowercase.
token-map	Maps a string through a mapping table.
token-parse-dn	Parses or converts a DN.
token-replace-all	Replaces all instances of a substring within a string.
token-replace-first	Replaces a single instance of a substring with a string.
token-split	Splits a string into a node set.
token-substring	Substring of a string.
token-upper-case	Converts a string to uppercase.
token-xml-parse	Parses XML.
token-xml-serialize	Serializes XML.

6 DirXML Entitlements DTD

Conceptually, a DirXML entitlement is a named flag that causes an Identity Manager driver configuration to perform some arbitrary action that is usually related to granting access to some resource in a connected system. Entitlements (as embodied in Role-based Entitlements) have thus far been used for three basic actions:

- Creating and deleting or disabling a connected-system account.
- Adding/removing connected-system accounts group memberships.
- Adding/setting attribute values to connected-system accounts.

An entitlement is embodied in an eDirectory DirXML-Entitlement object, which is contained by a DirXML-Driver object. The containment of the DirXML-Entitlement object establishes the correspondence between the entitlement and the implementing Identity Manager driver configuration. The DirXML-Entitlement object's name is the name of the entitlement. The XmlData attribute of the DirXML-Entitlement object contains an XML document whose root element is <entitlement>.

An entitlement is granted to and revoked from an eDirectory object via the addition of the auxiliary class DirXML-EntitlementRecipient and the associated DirXML-EntitlementRef attribute to the eDirectory object. The DirXML-EntitlementRef attribute is of SYN_PATH syntax and is “write-managed”. The “volume” (or DN) portion of the path syntax value refers to the DirXML-Entitlement object. Because the attribute is write-managed, the agent setting the DirXML-EntitlementRef attribute value on an eDirectory object must have write access to the DirXML-EntitlementRef attribute on the object that is being written to and must also have write access to the ACL attribute on the DirXML-Entitlement object that is referred to by the DN portion of the DirXML-EntitlementRef value. The “path” (or string) portion of the DirXML-EntitlementRef attribute contains an XML document whose root element is <ref>. The “namespace” (or integer) portion of the DirXML-EntitlementRef attribute is used as a bitmask to hold a set of flags. Bit 0 of the 32-bit integer is used for this flag value and is known as the state bit. 0 means revoked, 1 means granted. Bit 1 is used to flag a granted entitlement that is the result of the upgrade process and is known as the upgrade bit. 1 means that the entitlement was previously granted in the legacy format and is therefore not a change in the entitlement state. Bits 2-31 are reserved for future use.

After the entitlement action (grant or revocation) has been completed (successfully or not) by the Identity Manager driver configuration, a result is written to the eDirectory object using the DirXML-EntitlementResult attribute. DirXML-EntitlementResult is a multi-valued SYN_OCTET_STRING containing an XML document whose root element is <result>.

Since an entitlement is only a flag that signals an Identity Manager driver to grant some arbitrary resource, in order for the grant or revoke of an entitlement to actual have any effect, there must be policies on the driver that handle the actual granting or revoking of access to the resource in the connected application. DirXML Script contains explicit support for implementing entitlement policies. The <if-entitlement> condition is used to determine if a given entitlement has been granted or is changing. The <token-entitlement>, <token-added-entitlement>, and <token-removed-entitlement> tokens are used to get a list of the granted or revoked entitlements. The <do-implement-entitlement> action is used to mark policy actions that implement entitlements so that the results of the entitlement can be automatically logged to DirXML-EntitlementResult. The entitlement tokens

return a nodeset containing 0 or more <entitlement-impl> elements that can be used to get information about the entitlements and can be passed as an arguments to <do-implement-entitlement>.

See "DirXML Entitlements DTD Elements" on page 494 for a list of all of the elements in the DirXML Elements DTD.

6.1 DirXML Entitlements DTD Elements

Element	Description
description	Queries the result of the description.
display-name	Queries the result of the display name.
dn	Entitlement object DN.
ent-value	Queries the result value.
entitlement	Entitlement definition.
entitlement-impl	Entitlement implementation marker.
id	Entitlement granting agent correlation ID.
item	Cached entitlement query result item.
item-description	Cached entitlement query result item description.
item-display-name	Cached entitlement query result item display name.
item-value	Cached entitlement query result item value.
items	Cached entitlement query results.
msg	Entitlement result status message.
param	Entitlement parameter value.
query-app	Query definition for legal values of the entitlement.
query-xml	XDS query.
ref	Entitlement reference.
result	Entitlement result.
result-set	Interpretation of query results.
src	Entitlement granting agent type.
state	Entitlement state.
status	Entitlement result status level.
timestamp	Entitlement result time stamp.
token-association	Uses association value from the query result.

Element	Description
token-attr	Use attribute value from the query result.
token-src-dn	Use src-dn value from query result.
value	Enumerated value.
values	Legal values of the entitlement.

description

Specifies how to obtain the description for each of the XDS instance element returned from an XDS query used to dynamically obtain a list of possible values from the connected application for the enclosing <entitlement>.

Example

See <entitlement>.

Allowed Content

Element	Description
token-association	Use association value from the query result.
token-src-dn	Use association value from the query result.
token-attr	Use attribute value from the query result.

Content Rule

(token-association | token-src-dn | token-attr)

Parent Elements

Element	Description
result-set	Interpretation of the query results.

display-name

Specifies how to obtain a display-name for each of the XDS instance element returned from an XDS query used to dynamically obtain a list of possible values from the connected application for the enclosing <entitlement>.

Example

See <entitlement>.

Allowed Content

Element	Description
token-association	Use association value from the query result.
token-src-dn	Use src-dn value from the query result.
token-attr	Use attribute value from the query result.

Attributes

None

Content Rule

(token-attr | token-src-dn | token-association)

Parent Elements

Element	Description
result-set	Interpretation of the query results.

dn

Contains the LDAP format DN of the DirXML-Entitlement object referenced by the enclosing <result>.

Example

See <result>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
result-set	Interpretation of the query results.

ent-value

Specifies how to obtain the actual parameter value for each of the XDS instance element returned from an XDS query used to dynamically obtain a list of possible values from the connected application for the enclosing <entitlement>.

Example

See <entitlement>.

Allowed Content

Element	Description
token-association	Use association value from the query result.
token-src-dn	Use src-dn value from the query result.
token-attr	Use attribute value from the query result.

Attributes

None

Content Rule

(token-association | token-src-dn | token-attr)

Parent Elements

Element	Description
result-set	Interpretation of the query results.

entitlement

An entitlement definition that is stored in the XmlData attribute of a DirXML-Entitlement object. The actual name of the entitlement comes from the RDN of the DirXML-Entitlement object. The entitlement definition defines a display name and description for the entitlement for use in UI agents, and can define whether or not an entitlement requires a single parameter value. If a parameter value is required, then it is possible to specify a list of possible values for that parameter value, or a query that can be issued to the associated application in order to get a list of possible parameters values.

Example

```
<entitlement conflict-resolution="union" description="Recipient is entitled to an
account in the connected application" display-name="User Account"/>

<entitlement conflict-resolution="union"
    description="Groups that the recipient is entitled to belong to in the
connected application" display-name="Group Membership">
  <values>
    <query-app>
      <query-xml>
        <nds dtd-version="2.0">
          <input>
            <query class-name="Group" scope="subtree">
              <search-class class-name="Group"/>
              <read-attr attr-name="Description"/>
            </query>
          </input>
        </nds>
      </query-xml>
      <result-set>
        <display-name>
          <token-src-dn/>
        </display-name>
        <description>
          <token-attr attr-name="Description"/>
        </description>
        <ent-value>
          <token-association/>
        </ent-value>
      </result-set>
    </query-app>
  </values>
</entitlement>

<entitlement conflict-resolution="priority" description="The Musical Instrument
played by the recipient" display-name="Musical Instrument">
  <values multi-valued="false">
    <value>Trumpet</value>
    <value>Clarinet</value>
    <value>Trombone</value>
    <value>Flute</value>
    <value>Violin</value>
  </values>
</entitlement>
```

Allowed Content

Element	Description
values	Legal values of the entitlement.

Attributes

Attribute	Possible Values	Default Value
conflict-resolution	priority | union The conflict resolution method to be used by the Entitlements driver when the entitlement is used in conjunction with Role-based Entitlements and is granted via more than one role. union: The parameter values of all the granting roles are granted to the recipient. priority: Only the parameter values of the granting role with the highest priority are granted to the recipient.	priority
description	CDATA The description for the entitlement that should be displayed by a UI agent.	#REQUIRED
display-name	CDATA The name for the entitlement that should be displayed by a UI agent.	#REQUIRED

Content Rule

(values ?)

Parent Elements

None

entitlement-impl

Represents a granted or revoked entitlement within DirXML Script.

Example

```
<entitlement-impl id="xxx" src="AF" state="1" name="Group" src-
dn="\MYTREE\Novell\Users\Fred" src-entry-id="65535">
   cn=Managers,o=People
</entitlement-impl>
```

Allowed Content

#PCDATA

Attributes

Attribute	Possible Values	Default Value
id	CDATA The ID provided by the granting agent.	#IMPLIED
name	CDATA Name of the entitlement.	#REQUIRED
src	CDATA The granting agent.	#REQUIRED
src-dn	CDATA The eDirectory DN of the entitlement recipient in slash format.	#REQUIRED
src-entry-id	CDATA The eDirectory entry ID of the entitlement recipient in slash format	#IMPLIED
state	0 | 1 The grant/revocation state 1 - granted 0 - revoked	#REQUIRED

Content Rule

(#PCDATA)

Parent Elements

None

id

An ID provided to a granted or revoked entitlement that is provided by the granting/revoking agent and is carried forward on the result of the granted/revoked entitlement. The meaning of the id is transparent to the entitlements system and is primarily intended to be used by the granting/revoking agent to correlate the entitlement results with the original grant or revocation.

Example

See <ref>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
ref	Entitlement reference.
result	Entitlement result.

item

A cached query result item of a query used to dynamically discover the possible parameter values for an entitlement.

Example

See <items>.

Allowed Content

Element	Description
item-display-name	Cached entitlement query result item display name.
item-description	Cached entitlement query result item description.
item-value	Cached entitlement query result item value.

Attributes

None

Content Rule

(item-display-name ? , item-description ? , item-value)

Parent Elements

Element	Description
items	Cached entitlement query results.

item-description

The description of a cached query result item of a query used to dynamically discover the possible parameter values for an entitlement.

Example

See <items>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
item	Cached entitlement query result item.

item-display-name

The display name of a cached query result item of a query used to dynamically discover the possible parameter values for an entitlement.

Example

See <items>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
item	Cached entitlement query result item.

item-value

The parameter value of a cached query result item of a query used to dynamically discover the possible parameter values for an entitlement.

Example

See <items>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
item	Cached entitlement query result item.

items

Holds the cached query result items of a query used to dynamically discover the possible parameter values for an entitlement. It is stored in the DirXML-SPCacheExternalQuery attribute of a DirXML-Entitlement object.

Example

```
<items>
 <item>
  <item-display-name>cn=Managers,o=People</display-name>
  <item-description>Managers</description>
  <item-value>cn=Managers,o=People</value >
 </item>
 <item>
  <item-display-name>cn=Contractors,o=People</display-name>
  <item-description>Contractors</description>
  <item-value>cn=Contractors,o=People</value >
 </item>
</items>
```

Allowed Content

Element	Description
item	Cached entitlement query result item.

Attributes

None

Content Rule

(item *)

Parent Elements

None

msg

Contains the status message of the entitlement grant/revocation referenced by the enclosing <result>.

Example

See <result>.

Allowed Content

ANY

Attributes

None

Content Rule

ANY

Parent Elements

Element	Description
result	Entitlement result.

param

The parameter value of the entitlement as represented in a granted or revoked entitlement, and entitlement associated with a an RBE policy, or the result of granting/revoking an entitlement.

Example

See <ref>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
ref	Entitlement reference.
result	Entitlement result.

query-app

If present, a set of query-apps specifies an XDS query that can be used that can be used to dynamically obtain a list of possible values from the connected application for the enclosing <entitlement>. It also specifies how to obtain a display-name, description, and actual parameter value for each of the XDS instance elements returned from the query.

Example

See <entitlement>.

Allowed Content

Element	Description
query-xml	XDS query
result-set	Interpretation of query results

Attributes

None

Content Rule

(query-xml , result-set)

Parent Elements

Element	Description
values	Legal values of the entitlement.

query-xml

Specifies an XDS query that can be used that can be used to dynamically obtain a list of possible values from the connected application for the enclosing <entitlement>.

Example

See <entitlement>.

Allowed Content

ANY

Attributes

None

Content Rule

ANY

Parent Elements

Element	Description
query-app	Query definition for legal values of the entitlement.

ref

The value of the “path” (or string) portion of the DirXML-EntitlementRef attribute. When used on a DirXML-EntitlementRecipient it represents a granted or revoked entitlement and contains information about the granting/revoking agent as well as the parameter value if the entitlement requires one. When used on a DirXML-SharedProfile (that is, an RBE policy or role), it is only used to provide the parameter value that will be granted by the role.

Example

```
<ref>
  <src>RBE</src>
  <id>{26dfb70f-0371-4fe2-a67f-bc101101e5d7}</id>
  <param>cn=Managers,o=People</param>
</ref>
```

Allowed Content

Element	Description
src	Entitlement granting agent type.
id	Entitlement granting agent correlation ID.
param	Entitlement parameter value.

Attributes

None

Content Rule

(src ? , id ? , param ?)

Parent Elements

None

result

Written as a value of the DirXML-EntitlementResult attribute of a DirXML-EntitlementRecipient. It contains the results of granting or revoking an entitlement, as implemented by policy.

Example

```
<result>
  <dn>cn=Group,cn=LDAP Driver,cn=DriverSet,o=
novell</dn>
  <src>RBE</src>
  <id>{26dfb70f-0371-4fe2-a67f-bc101101e5d7}</id>
  <param>cn=Managers,o=People</param>
  <state>1</state>
  <status>error</status>
  <msg>Access denied</msg>
  <timestamp>1112101901523</timestamp>
</result>
```

Allowed Content

Element	Description
dn	Entitlement object DN.
src	Entitlement granting agent type.
id	Entitlement granting agent correlation ID.
param	Entitlement parameter value.
state	Entitlement state.
status	Entitlement result status level.
msg	Entitlement result status message.
timestamp	Entitlement result time stamp.

Attributes

None

Content Rule

(dn , src , id ? , param ? , state , status , msg ? , timestamp)

Parent Elements

None

result-set

Specifies how to obtain a display-name, description, and actual parameter value for each of the XDS instance element returned from an XDS query used to dynamically obtain a list of possible values from the connected application for the enclosing <entitlement>.

Example

See <entitlement>.

Allowed Content

Element	Description
display-name	Query the result of the display name.
description	Query the result of the description.
ent-value	Query the result value.

Attributes

None

Content Rule

(display-name , description , ent-value)

Parent Elements

Element	Description
query-app	Query the definition for legal values of the entitlement.

src

Used to identify the granting or revoking agent for an granted/revoked entitlement. Well know values are RBE when the agent is the Role Based Entitlements system, or AF when the granting agent is the Approval Flow system. Other agents that grant entitlements should provide their own unique identifier.

Example

See <ref>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
ref	Entitlement reference.
result	Entitlement result.

state

Contains the state (1=granted, 0=revoked) of the entitlement referenced by the enclosing <result>.

Example

See <result>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
result	Entitlement result.

status

Contains the status (success, error, warning, retry, fatal) of the entitlement grant/revocation referenced by the enclosing <result>.

Example

See <result>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
result	Entitlement result

timestamp

Contains the time stamp of the entitlement grant/revocation referenced by the enclosing <result>. The time stamp is an integer that represents the number of milliseconds since midnight January 1, 1970.

Example

See <result>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
result	Entitlement result.

token-association

Specifies that the value of the association key of the XDS instance element should be used as the value of the enclosing <display-name>, <description>, or <ent-value>.

Example

See <entitlement>.

Allowed Content

EMPTY

Attributes

None

Content Rule

Empty

Parent Elements

Element	Description
description	Query the result of the description.
display-name	Query result display name.
ent-value	Query result value.

token-attr

Specifies that the first value of the named attribute from the XDS instance element should be used as the value of the enclosing <display-name>, <description>, or <ent-value>.

Example

See <entitlement>.

Allowed Content

EMPTY

Attributes

Attribute	Possible Values	Default Value
attr-name	CDATA Name of the attribute.	#REQUIRED

Content Rule

(#PCDATA)

Parent Elements

Element	Description
description	Query the result of the description.
display-name	Query result display name.
ent-value	Query result value.

token-src-dn

Specifies that the value of the src-dn attribute of the XDS instance element should be used as the value of the enclosing <display-name>, <description>, or <ent-value>.

Example

See <entitlement>.

Allowed Content

EMPTY

Attributes

None

Content Declaration

Empty

Parent Elements

Element	Description
description	Query the result of the description.
display-name	Query result display name.
ent-value	Query result value.

value

If present, a set of <value> elements specifies a static list of possible values for the enclosing <entitlement>.

Example

See <entitlement>.

Allowed Content

#PCDATA

Attributes

None

Content Rule

(#PCDATA)

Parent Elements

Element	Description
values	Legal values of the entitlement.

values

If present, <values> specifies that the enclosing <entitlement> requires a parameter value. If multi-valued="false" then the entitlement can only be granted to a given recipient with single value at a time, otherwise the entitlement can be granted to the same recipient more than once, each with a different value. <values> might also specify a list of possible values for the parameter value, or a query that can be issued to the associated application in order to get a list of possible parameters values.

Example

See <entitlement>.

Allowed Content

Element	Description
query-app	Query definition for legal values of the entitlement.
value	Enumerated value.

Attributes

Attribute	Possible Values	Default Value
multi-valued	True | False	True
	Multi-valued flag.	

Content Rule

(query-app | value +) ?

Parent Elements

Element	Description
entitlement	Entitlement definition.

7 Jobs DTD

The Identity Manager Job Scheduler lets you create and schedule jobs that you want to perform on Identity Manager servers. This section introduces the XML structure of the job objects used by the Job Scheduler, and includes the following topics:

7.1 Jobs XML

The XML that defines a job consists of two main parts: a `<job-definition>` section and zero or more `<xliff>` sections that provide localized strings for the `<job-definition>`. The two parts are aggregated in order to ease storage on the job definition object. The Jobs DTD is as follows:

```
<!ENTITY % Job-Type "java | script">
<!ENTITY % Result-Type "intermediate | final">
<!ENTITY % Result-Level "success | warning | error | aborted">
<!ENTITY % Boolean "true | false">
<!ELEMENT job-aggregation (job-definition, xliff*)>
<!ELEMENT job-definition (description, containment*, java-class, configuration-
values?, result-processing*>
<!ATTLIST job-definition
  display-name CDATA #REQUIRED
  type (%Job-Type) #REQUIRED
  schedule CDATA #IMPLIED
  scope-required (%Boolean) "false"
  disabled (%Boolean) "false"
  auto-delete (%Boolean) "false"
>
<!ELEMENT description (#PCDATA) >
<!ELEMENT containment (#PCDATA) >
<!ELEMENT java-class (#PCDATA)>
<!ELEMENT result-processing (audit | email)+ >
<!ATTLIST result-processing
  type (%Result-Type) "final"
>
<!ELEMENT audit NONE>
<!ATTLIST audit
  on-level (%Result-Level) #REQUIRED
>
<!ELEMENT email (to+, cc*, bcc*, reply-to+)>
<!ATTLIST email
  on-level (%Result-Level) #REQUIRED
  encoding CDATA #IMPLIED
>
<!ELEMENT to (#PCDATA) >
<!ELEMENT cc (#PCDATA) >
<!ELEMENT bcc (#PCDATA) >
<!ELEMENT reply-to (#PCDATA) >
```

The Jobs XML makes use of the following elements:

audit

Indicates that job result notification will be issued through the Novell Audit system.

Attributes

Attribute	Possible Values	Default Value
on-level	success | warning | error | aborted Specifies the result level for which this audit notification applies. You can include an audit elements for each result level that you want to generate a notification.	#REQUIRED

Parent Elements

result-processing

bcc

Specifies a list of email addresses to place in the blind copy (*BCC*) field when sending a results notification email.

Attributes

None

Parent Elements

email

cc

Specifies a list of email addresses to place in the carbon copy (*CC*)) field when sending a results notification email.

Attributes

None

Parent Elements

email

containment

Zero or more containment elements describe any restrictions to the eDirectory objects that can contain the `DirXML-Job` object representing the job. If no containment element appears then the `DirXML-Job` object may be contained by either a `DirXML-DriverSet` or `DirXML-Driver` object.

Attributes

None

Parent Elements

job-definition

description

The description element contains a human-readable description of the job.

Attributes

None

Parent Elements

job-definition

email

Indicates that job result notifications are issued via an email message. The email message is constructed from an email template specified by the `DirXML-EmailTemplates` attribute on the `DirXML-Job` object. The email recipients are specified in the email element's child elements: `to`, `cc`, and `bcc`. The email's reply-to address is specified by the email element's child reply-to element.

Attributes

Attribute	Possible Values	Default Value
on-level	success | warning | error | aborted Specifies the result level for which this email notification applies. You can include an audit elements for each result level that you want to generate a notification.	#REQUIRED
encoding	CDATA Specifies the encoding method to use with the email message.	#IMPLIED

Parent Elements

result-processing

java-class

Contains the fully-qualified name of the java class that implements the job. There must be exactly one java-class element per job.

Attributes

None

Parent Elements

job-definition

job-aggregation

Functions as the container object for a job and aggregates the job-definition information and any xliff data used for language translation.

Attributes

None

Parent Elements

None

job-definition

Contains XML attributes and elements which define much of the job.

Attributes

Attribute	Possible Values	Default Value
display-name	CDATA Specifies the job name displayed in the Job Scheduler UI.	#REQUIRED
type	java Specifies the language used to write the job. The only supported option is java.	#REQUIRED
schedule	CDATA Specifies the schedule used to run the job. This is configured in the Job Scheduler UI.	#IMPLIED
scope-required	true | false Specifies if a job scope is necessary. The job scope defines the Identity Vault objects that the job works with, and is determined by the job developer.	False
disabled	true | false Specifies, when true, that the job is not available to run.	False
auto-delete	true | false Indicates, when true, that the job is a one-time job that should be deleted after it runs.	False

Parent Elements

job-aggregation

reply-to

Specifies a Reply To email address to place in the *From* field when sending a results notification email.

Attributes

None

Parent Elements

email

result-processing

Describes how the job results (both intermediate and final) are reported. There can be zero or more result-processing elements. A result-processing element describes how notification of job results is made; and contains one or more audit or email elements. Each result-processing element applies to either "final" or "intermediate" results, based on the value of the element's type attribute.

Attributes

Attribute	Possible Values	Default Value
type	intermediate | final Specifies whether the results-processing element applies to intermediate results that occur during job processing, or the final results when the job completes.	final

Parent Elements

job-definition

to

Specifies a list of email addresses to place in the *To* field when sending a results notification email.

Attributes

None

Parent Elements

email

xliff

Provides localized strings for use by the job-definition object. The xliff format is an open standard. For more information about xliff, see the XLIFF Web site (http://www.oasis-open.org/committees/xliff).

Parent Elements

job-aggregation

7.2 Example Job XML

Example 1

```
<job-aggregation>
<job-definition
  type="java"
  schedule="0 2 * * 1-5"
>
  <containment>DirXML-Driver</containment>
  <java-class>com.novell.nds.dirxml.jobs.builtin.PasswordGeneration</java-class>
  <configuration-values>
    <definitions>
      <definition
        type="dn-ref"
        name="password-policy"
        display-name="Password policy object"
        attr-name="DirXML-PasswordPolicyRef"
        aux-class-name="DirXML-PasswordGenAttrs"
      >
        <value/>
      </definition>
    </definitions>
  </configuration-values>
</job-definition>
</job-aggregation>
```

Example 2

```
<job-aggregation>
<job-definition
  type="java"
  schedule="0 2 * * 1-5"
>
  <containment>DirXML-Driver</containment>
  <java-class>com.novell.nds.dirxml.jobs.builtin.SubSubmitter</java-class>
  <configuration-values>
    <definitions>
      <definition
        type="string"
        name="doc"
        display-name="Document to submit"
        multiline="true"
      >
        <value>
          <nds>
            <input>
```

```
              <query scope="subtree">
                <association>{123456}</association>
                <search-class class-name="User"/>
                <read-attr/>
              </query>
            </input>
          </nds>
        </value>
      </definition>
    </definitions>
  </configuration-values>
</job-definition>
</job-aggregation>
```

8 Global Configuration Values

Global configuration values (GCVs) are values that can be used by the driver to control functionality. GCVs are defined on the driver or on the driver set. Driver set GCVs can be used by all drivers in the driver set. Driver GCVs can be used only by the driver on which they are defined. GCVs contain definitions related to the XML representation of global configuration values.

The precedence order in which the GCV definitions are loaded is:

1. GCV definitions on the driver object.
2. GCV definitions in the DirXML-GlobalConfigDef objects linked to the driver.
3. GCV definitions on the Driverset object.
4. GCV definitions in the DirXML- GlobalConfigDef objects linked to the Driverset.

GCVs in a driver configuration provide one or more typed values which are presented to the user (typically an administrator) by a user interface agent such as iManager or Designer, hereafter referred to as the Agent. The user can set values via the Agent. These values are used to make decisions in the driver configuration's policies.

Control values are described syntactically by an XML vocabulary. The vocabulary consists of structural elements, definition elements, reference elements, and control elements. Structural elements are essentially housekeeping units that serve to fulfill XML syntax requirements. Definition elements define the actual control values with their data types and other information. Reference elements are used to refer to data outside the definition document. Control elements affect how the Agent presents the values to the user. The following is an example of a control value XML definition:

```
<configuration-values>
<definitions>
  <definition
   display-name="Send email on failure"
   name="send-email"
   type="boolean">
 <value>true </value>
 </definition>
</definitions>
</configuration-values>
```

The control value definitions might be standalone XML documents or embedded in other XML documents depending on usage. For example, GCVs are defined in stand-alone documents found in stream-syntax attributes on DirXML-DriverSet and DirXML-Driver objects while Shim Parameters are embedded in a <driver-config> XML document. GCV definitions can also be present in the DirXML-Config driver object (resource object) or the driver set object. In past GCV's have been stored this way; however, with Identity Manager 4.0, GCVs are stored in the DirXML-Config objects. The DirXMl-ShimConfigInfo attribute uses the same DTD to define driver Configuration settings.

An Agent that presents control values to a user is responsible for parsing the XML definition, presenting the values in a meaningful way, allowing the user to make allowable changes to the values, and finally for storing the XML definition with updated values. Some Agents might also

provide for defining CVs in some cases (e.g., Designer as part of editing a driver configuration or as part of creating a Job definition). There exist Java classes in Identity Manager that can be used by Agents to assist with the parsing, correctness checking, and serialization of CV definition documents.

8.1 Common XML Constructs

The following XML attributes are required (or have a default value) on a `<definition>` element and are common to all types:

Attribute	Possible Values	Default Value
name	NMTOKEN	#REQUIRED
	The configuration value name. Must be unique among all descendant `<definition>` elements found under the `<definitions>` element.	
display-name	CDATA	String
	The string presented to the user to identify the value.	
display-name-ref	Display names or descriptions that have localization refs.	#IMPLIED
dn-type	“ldap”, "slash”, “qualifiedslash”, “dot”, “qualifieddot”, “custom”	#IMPLIED
	Specifies the syntax allowed for the value. Agents must enforce this when the user enters the value.	
hide	Boolean	#IMPLIED
mandatory	"true" or "false"	
	If "true" the Agent must require the user to supply a value for the CV.	
type	"string", "boolean", "integer", "real", "dn", "enum", "list", "passwordref", "dn-ref"	#IMPLIED
	The value type. If not present, the type defaults to string.	

The following elements are content of a <definition> element and are common to all types except the "gcvref" type:

Name	Explanation
value	Contains the current value of the CV. Actual allowed content is defined by the individual types.
description	A string presented to the user to explain the value in greater detail than is present in the `display-name` attribute. Note that the description text might contain line breaks that must be honored when displaying the description.
	Description may have a description-ref attribute that is part of the localization scheme used with ECVs.

8.2 Value Types

Each configuration value has an associated data type. If the data type is not specified on the value's XML `<definition>` element, the data type defaults to “string”.

The engine enforces configuration value data typing. If a configuration value definition has an invalid data type, the operation depending on the configuration value definition fails (for example, starting a driver or job).

8.2.1 string

Values of data type “string” have a value that is a sequence of Unicode characters. Any valid XML character that is allowed in XML PCDATA is allowed in the value of a string value. String values might be empty (such as, the string value might have zero length). A string value might have a multiline attribute on the `<definition>` element. If multiline has a value of "true", the Agent should present the value honoring any embedded end-of-line characters and allow the user to specify line breaks.

8.2.2 boolean

Values of data type “boolean” have a value that is either “true” or “false”. Boolean values might not have empty values. The content of the value element must match the following production:

Boolean ::= 'true' | 'false'

Example:

```
<definition
     type="boolean"
     name="perform-magic"
     display-name="Do what I mean, not what I say" >
     <value>true</value>
     <description>This is an example of a boolean value</description>
</definition>
```

8.2.3 integer

Values of data type “integer” have a value that is one or more Unicode characters that match the XML specification pattern Digit. Integer values might optionally be preceded by a `'-'` (hyphen) character indicating negation. The minimum and maximum integer values are those that can be

represented in a signed, twos-complement 32-bit value as in the Java language `int` type. Integer values may be restricted in range. A lower bound, an upper bound, or both may be specified. The content of the value element must match the following pattern:

Integer ::= '-'? Digit+

Digit ::= '1'|'2'|'3'|'4'|'5'|'6'|'7'|'8'|'9'

Example:

```
     <definition
           type="integer"
           name="how-much"
           display-name="Enter the amount to send to Perin">
           <value>0</value>
            <description>This amount will be deducted automatically from your
credit card account and sent to Perin.</description>
    </definition>
    <definition
         type="integer"
         name="port-number"
         display-name="Enter the TCP port to use"
         range-lo="1024"
         range-hi="65535">
         <value>8080</value>
          <description>This is the TCP port that will be used for connecting to the
application.</description>
     </definition>
     <definition
           type="integer"
           name="timeout"
           display-name="Timeout value in seconds"
           range-lo="1">
           <value>10</value>
    <description>This is the timeout value in seconds; the driver will wait up
to this number of seconds for a connection.</description>
    </definition>
```

8.2.4 real

Values of data type “real” have a value that describes a real (or floating-point) number. The total allowable range of values for a real value are those that can be represented by a 64-bit double-precision floating point value as in the Java language double type. Real values might be restricted in range. A lower bound, an upper bound, or both might be specified. The content of the value element must match the following production:

Real ::= Integer | ('-'? Digit+ '.' Digit+)

Example:

```
        <definition
             type="real"
             name="multiplier"
             display-name="Factor by which to multiply collective IQ"
             range-hi="1.0">
             <value>0.6</value>
            <description>When one or more managers walk into a room, the collective
IQ is affected. This value is used to calculate the approximate resulting
collective IQ.</description>
         </definition>
```

8.2.5 dn

Values of data type “dn” have a value that must be a syntactically-legal distinguished name. DN values must be further qualified with a dn-type attribute that specifies the DN syntax. A DN value might have an empty value. If dn-space="dirxml", an object selection button should be presented.

Examples:

```
          <definition
              type="dn"
              name="user-account"
               display-name="Enter the user account to be used for authentication"
              dn-type="ldap">

              <value>cn=admin,ou=ITDept,o=TheCompany</value>
               <description>This value specifies the DN of a user account used to
authenticate to the LDAP directory.</description>
              </definition>
              <definition type="dn" name="exchange-account"
              display-name="Enter the DN of the Exchange user"
              dn-type="custom"
              dn-delims="10./+=*\">

              <value>CN=Admin+UID=1056/OU=Mayberry</value>
         </definition>
```

8.2.6 enum

Values of data type “enum” have a value that is one of a defined set of strings. Each defined string must conform to the string production.

Example:

```
       <definition
          type="enum"
          name="magical-features"
          display-name="Choose the magical feature set">

          <value>full</value>
           <enum-choice display-name="Full-on, magical stuff">full</enum-choice>
           <enum-choice display-name="Partial magic, but no rabbits">partial</
enumchoice>
            <enum-choice display-name="No magic, just boring normal life">none</
enumchoice>
                </definition>
                <definition
                     display-name="xlfid(submit-method)Scheduled action"
                     name="action"
                     type="enum">

   <enum-choice display-name="xlfid(start)Start the driver">start</enumchoice>
   <enum-choice display-name="xlfid(stop)Stop the driver">stop</enum-choice>
   <enum-choice display-name="xlfid(toggle)Toggle the driver">toggle</enumchoice>
   <value>start</value>
                  </definition>
```

8.2.7 list

Values of data type “list” have a value that is an ordered list of zero or more strings. The list value also specifies a delimiter character that is used when outputting the list value in a non-structured location (e.g., as a single string consisting of each individual string in the list separated by the delimiter character).

The content of each <item> element must match the following production:

String ::= Char*

The list items can be empty or duplicate other list items. The order of list items is significant. The Agent must preserve all characters in a list item exactly as specified by the user. This includes whitespace. The Agent serializing the definition XML must therefore specify an xml:space="preserve" attribute on the value element. It might also be necessary to use character entity references for some whitespace based on the XML rules for parsers:

- Whitespace Handling (http://www.w3.org/TR/REC-xml/#sec-white-space)
- End-of-line Handling (http://www.w3.org/TR/REC-xml/#sec-line-ends)

Example:

```
<definition
display-name="List o' things"
item-separator=";"
name="ingredients"
type="list">

<value>
<item>Snips</item>
<item>Snails</item>
<item>Puppy Dog Tails</item>
<item>Sugar</item>
<item>Spice</item>
<item>Everything Nice</item>
</value>
</definition>
```

8.2.8 structured

Values of data type “structured” have two fundamental parts: a template that defines a set of simple types (string, enum, etc.), and zero or more instances that contain the actual values of the structured control value. In a sense structured values are similar to structures in the C programming language. The structured value also specifies delimiter strings that are used when outputting the value as a single string. The delimiters are used for string representation, but in a nodeset context, you get the <instance> node tree fragments. The string representation of the value contains all the components of the structured value separated by the delimiter, for example, <ipaddress>, <port>, and so on. However, the xml representation of the structured GCV has instances that contain the actual Global Definition values.

Example: Suggested Agent presentation for nested control value presentation with buttons for adding and removing instances. The example allows for template creation and editing.

```
<definition
display-name="Servers list"
value-separator=";"
instance-separator="&#10;"
name="servers"
type="structured">

<template>
<definition name="host" display-name="Host" type="string">
<value/>
</definition>
<definition name="port" display-name="TCP port" type="integer" rangelo="
1" range-hi="65535">
<value>1</value>
</definition>
</template>
<value>
<instance>
<definition name="host" display-name="Host" type="string">
<value>192.168.0.1</value>
</definition>
<definition name="port" display-name="TCP port" type="integer"
range-lo="1" range-hi="65535">
<value>8028</value>
</definition>
</instance>
<instance>
<definition name="host" display-name="Host" type="string">
<value>10.0.0.1</value>
</definition>
<definition name="port" display-name="TCP port" type="integer"
range-lo="1" range-hi="65535">
<value>8028</value>
</definition>
</instance>
</value>
</definition>
```

8.2.9 password-ref

Values of data type “password-ref” have a value that is the key value of a named password. Named password key values might be any non-empty sequence of Unicode characters. Because of the representation of CV definitions in XML, the actual set of Unicode characters allowed are those that are legal in XML PCCHAR data. The Agent presenting the password-ref value is responsible for setting any user-entered password value to the named password. If the referenced named password does not exist then the Agent must create it. The user-entered value must not be placed in the XML definition.

Example for Agent presentation of password field, associated confirmation field, and associated clear value button.

```
<definition
  type="password-ref"
  name="account-pwd"
  display-name="Password for the authentication account">

  <value>auth-acct-pwd</value>
</definition>
```

8.2.10 dn-ref

The dn-ref data type does not have a a value of its own. Instead, it assumes any value obtained from a DN-syntax eDirectory attribute on the object containing the value definition. The information about the eDirectory attribute is specified by XML attributes on the <definition> element.

The following example shows an Agent presentation of an entry field with the associated object selection button.

```
<definition
       type="dn-ref"
       name="pwd-policy"
       display-name="Password Policy object used for password generation"
       attr-name="nspmPasswordPolicyDN"
       aux-class-name="DirXML-PasswordGeneration">
    <target-class>nspmPasswordPolicy</target-class>
 <value>DirXML-PasswordPolicy.Password Policies.Security</value>
</definition>
```

8.2.11 gcv-ref

A GCV reference value does not have a value of its own and is not defined with a <definition> element. Instead, a value is obtained from and stored to a GCV in an enclosing logical structure. For example, a gcv-ref value in shim parameters refers to a GCV defined for the driver. The GCV is specified by the name attribute on the <gcv-ref> element.

The following example shows an Agent presentation that depends on the referenced GCV:

```
<gcv-ref name="placement-base"/>

<definition display-name="Placement base container" name="placement-base"
type="string">
               <description>This is  the placement base container</description>
              <value></value>
        </definition>
```

8.3 GCV DTD

The XML that defines GCVs consists of a <definition> section. The GCV DTD is as follows:

```
<!ENTITY % Value-Type "string | boolean | integer | real | dn | enum | list |
password-ref | dn-ref">
<!ENTITY % Dn-Type "ldap | slash | qualified-slash | dot | qualified-dot | custom">
<!ENTITY % Dn-Space "dirxml | application">
<!ENTITY % Boolean "true | false">
<!ELEMENT configuration-values (definitions)>
<!ELEMENT definitions (definition | gcv-ref | header | group)*>
<!-- FIXIT: one value, one description -->
<!ELEMENT definition (value | description | enum-choice | target-class)*>
<!ATTLIST definition
  name NMTOKEN #REQUIRED
  display-name CDATA #REQUIRED
  type (%Value-Type;) "string"
  dn-type (%Dn-Type;) #IMPLIED
  dn-delims CDATA #IMPLIED
  dn-space (%Dn-Space;) #IMPLIED
  range-lo CDATA #IMPLIED
  range-hi CDATA #IMPLIED
  multiline CDATA "false"
  attr-name CDATA #IMPLIED
  aux-class-name CDATA #IMPLIED
  item-separator CDATA #IMPLIED
```

```
  display-name-ref CDATA #IMPLIED
  hide (%Boolean;) #IMPLIED
>
<!ELEMENT gcv-ref EMPTY>
<!ATTLIST gcv-ref name NMTOKEN #REQUIRED>
<!ELEMENT group ((definition | gcv-ref),(definition | gcv-ref | subordinates |
header | group)*)>
<!ELEMENT subordinates (group | definition | gcv-ref | header)*>
<!ATTLIST subordinates active-value CDATA #REQUIRED>
<!ELEMENT header EMPTY>
<!ATTLIST header display-name CDATA #REQUIRED>
<!ELEMENT value (#PCDATA | item)*>
<!ATTLIST value xml:space (preserve|default) #IMPLIED>
<!ELEMENT description (#PCDATA)>
<!ATTLIST description  description-ref CDATA #IMPLIED>
<!ELEMENT enum-choice (#PCDATA)>
<!ATTLIST enum-choice    display-name CDATA #REQUIRED>
<!ELEMENT item (#PCDATA)>
<!ATTLIST item xml:space (preserve) #FIXED "preserve">
<!ELEMENT target-class (#PCDATA)>
```

8.4 GCV DTD Elements

definition

Definition elements define the actual control values with their data types and other information.

Attributes

The following XML attributes are required (or have a default value) on a <definition> element and are common to all types:

Attribute	Possible Values	Default Value
attr-name	A non-empty CDATA value. The name of the eDirectory attribute containing the value. The attribute must be of syntax "Distinguished Name".	#IMPLIED
aux-class-name	A non-empty CDATA value. The name of the eDirectory auxiliary class that attaches the eDirectory attribute to the object containing the CV definition.	#IMPLIED
description	A string presented to the user to explain the value in greater detail than is present in the `display-name` attribute. Note that the description text might contain line breaks that must be honored when displaying the description. Description may have a description-ref attribute that is part of the localization scheme used with ECVs.	
display-name	CDATA The string presented to the user to identify the value.	"string"
display-name-ref	non-empty CDATA value Used with ECVs as part of the localization scheme.	#IMPLIED
dn-type	“ldap”, "slash”, “qualifiedslash”, “dot”, “qualifieddot”, “custom” Specifies the syntax allowed for the value. Agents must enforce this when the user enters the value.	#IMPLIED

Attribute	Possible Values	Default Value
dn-delims	A sequence of at least eight characters that specify the custom DN format delimiters. If dn-type = "custom" then dn-delims must contain the "delimiter set" for the custom DN syntax. The delimiter set is at least 8 characters as follows: ◆ '0' indicating untyped names or '1' indicating typed names. ◆ '0' or '1' indicating "do not" or "do" hex escape unmappable characters in DN string. ◆ Character used for relative DNs ◆ Character used to separate RDNs ◆ Delimiter in names ◆ Value indicator ◆ Wildcard character ◆ Escape character Any characters after the eighth are additional characters that must always be escaped in the DN syntax. Note that dn-delims is ignored if dn-type is equal to anything other than "custom".	#IMPLIED
dn-space	"dirxml" or "application" If the value is "dirxml" then the Agent should present an "object selector" button that allows an object to be selected from eDirectory.	#IMPLIED
hide	Boolean	#IMPLIED
item-separator	A non-empty CDATA value. The value is used as a separator between list item strings when the list value is output in a non-structured location (e.g., as part of a single string). For example, if the list items are "one", "two", and "three", and the item-separator value is ";", the following is the unstructured output: "one;two;three"	#IMPLIED
multiline	"true" or "false" If "true" then the Agent should present the value in a multi-line edit field and allow the user to specify line breaks. The content of the value element must match the following production: String ::= Char* The Agent must preserve all characters exactly as specified by the user. This includes whitespace. The Agent serializing the definition XML must therefore specify an xml:space="preserve" attribute on the value element. It may also be necessary to use character entity references for some whitespace based on the XML rules for parsers: Whitespace Handling End-of-line Handling	False

Attribute	Possible Values	Default Value
name	NMTOKEN	#REQUIRED
	The configuration value name. Must be unique among all descendant <definition> elements found under the <definitions> element.	
range-hi	A CDATA value that matches the syntax allowed for the value element content.	#IMPLIED
	Specifies the maximum integer or real value allowed. The Agent must enforce this when the user specifies the value.	
range-lo	A CDATA value that matches the syntax allowed for the value element content.	#IMPLIED
	Specifies the minimum integer or real value allowed. The Agent must enforce this when the user specifies the value.	
type	"string", "boolean", "integer", "real", "dn", "enum", "list", "passwordref", "dn-ref"	#IMPLIED
	The value type. If not present, the type defaults to "string".	
value	Contains the current value of the CV. Actual allowed content is defined by the individual types.	

Parent Elements

description

description

Specifies the GCV description.

Attributes

Attribute	Possible Values	Default Value
description ref	CDATA	#IMPLIED

Parent Elements

description

gcv-ref

A GCV reference value does not have a value of its own and is not defined with a `<definition>` element. Instead, any value is obtained from and stored to a GCV in an enclosing logical structure (e.g., a gcv-ref value in Shim Parameters refers to a GCV defined for the driver). The referred to GCV is specified by the name attribute on the `<gcv-ref>` element.

Attributes

Attribute	Possible Values	Default Value
gcv-ref name	A non-empty CDATA value. The name of the referenced GCV.	#REQUIRED
driver-param name	A string that matches the XML Name production. The name of the driver parameter element when constructing shim parameters. Normally, the name of the driver parameter used is the name of the referred-to GCV; if this attribute's value is non-empty then the attribute value is used as the name of the driver parameter.	string

Example:

```
<gcv-ref name="placement-base"/>
```

Parent Elements

definition

group

A <group> element instructs the user interface Agent to consider all content of the <group> element as related. The first element child of the group is the group leader and must be either a boolean or an enum type value (or a <gcv-ref> that refers to a boolean or an enum). Subsequent members of the group should be displayed as subordinate to the group leader.

If the group leader is a <gcv-ref>, the definition it refers to must be a boolean or an enum.

Attributes

Attribute	Possible Values	Default Value
definition	"boolean" or "enum" The <group> might contain any number of <definition> elements. If the first <group> child element is a <definition> element, the type attribute value must be boolean or enum.	
gcv-ref	The <group> may contain any number of <gcv-ref> elements. If the first <group> child is <gcv-ref> then the referenced GCV must be a boolean or enum.	
subordinates	Any number of <subordinates> elements may appear as children of a <group> element. However, a <subordinates> element may not be the first child of a <group> element.	
header	Any number of <header> elements may appear as children of a <group> element. However, a <header> element may not be the first child of a <group> element.	
group	Any number of <group> elements may appear as children of a <group> element. However, a <group> element may not be the first child of a <group> element.	#REQUIRED

Example:

```
<header display-name="TCP parameters"/>
  <group>
        <definition
          type="boolean"
          name="server-connect"
          display-name="Connect to remote server"
                        >
           <value>true</value>
         </definition>
        <definition
 type="string"
 name="host-name"
 display-name="Host name or IP address of server"
  >
       <value>192.168.0.1</value>
       </definition>
        <definition
          type="integer"
          name="port"
          display-name="HTTP port number of server"
          range-lo="1"
          range-hi="65535"
        >
       <value>80</value>
     </definition>
 </group>
```

Parent Elements

definition

subordinates

A <subordinates> element only appears as part of a group and delimits content that is conditionally displayed by the user interface Agent. A <subordinates> element may not be a group leader; instead, the group leader's value determines whether the <subordinates> element's content is displayed or not. The value that will cause the <subordinates> element's content to be displayed is determined by the value of the active-value attribute.

Attributes

Attribute	Possible Values	Default Value
active-value	A non-empty CDATA value. Specifies the value of the group leader definition that causes the Agent to display the contents of the <subordinates> element. The value must be a legal value for the group leader.	#REQUIRED

A group can have more than one subordinates element under it. Each subordinate has a different value for the active-value attribute.

Element Name	Explanation
definition	The <subordinates> element might contain any number of <definition> elements.
gcv-ref	The <subordinates> element might contain any number of <gcv-ref> elements.
header	The <subordinates> element might contain any number of <header> elements.
group	The <subordinates> element might contain any number of <group> elements.

Parent Elements

group

header

A header instructs the user interface Agent to display some sort of delimiter or header with the display-name text. Any number of `<header>` elements may appear as children of a <group> element. However, a `<header>` element might not be the first child of a `<group>` element.

Attributes

Attribute	Possible Values	Default Value
display-name	A CDATA value. The string that will be presented to the user.	#REQUIRED

Example for Agent presentation for larger type for display name, some sort of horizontal rule:

```
<header display-name="TCP parameters"/>
```

Example for Agent presentation for indent subordinates.

```
<header display-name="TCP parameters"/>
<group>
<definition
      type="boolean"
      name="server-connect"
      display-name="Connect to remote server">

      <value>true</value>
      </definition>
      <definition
      type="string"
      name="host-name"
      display-name="Host name or IP address of server">

      <value>192.168.0.1</value>
</definition>
<definition
      type="integer"
      name="port"
      display-name="HTTP port number of server"
      range-lo="1"
      range-hi="65535">

      <value>80</value>
</definition>
 </group>
```

Parent Elements

definition, group

value

The content of the value element must either be empty or be a sequence of characters that is syntactically valid for the DN syntax specified by the dn-type attribute on the `<definition>` element. The content of the value element must a string from one of the <enum-choice> elements. The value element content must not be empty. The content of the value element must be zero or more <item> elements. The content of the value element must be non-empty and is the key value (name) of a named password.

Attributes

Attribute	Possible Values	Default Value
xml:space	preserve|default	#IMPLIED

Parent Elements

definition

enum-choice

One or more <enum-choice> elements must be children of the <definition> element. Each <enum-choice> element has a display-name attribute that is used by the Agent to present the choice to the user. The content of an <enum-choice> element must conform to the following production: String ::= Char+. The content of an <enum-choice> element must not be empty and defines one of the allowable strings for the <value> element content.

Attributes

Attribute	Possible Values	Default Value
display-name		#REQUIRED

Parent Elements

definition

item

The content of the value element must be zero or more <item> elements. The content of each <item> element must match the following production:

String ::= Char*

Note that this means that list items may be empty. List items may duplicate other list items. The order of list items is significant. The Agent must preserve all characters in a list item exactly as specified by the user. This includes whitespace. The Agent serializing the definition XML must therefore specify an xml:space="preserve" attribute on the value element.

Attributes

Attribute	Possible Values	Default Value
xml:space="preserve"		#FIXED

Parent Elements

definition

target-class

One or more <target-class> elements can appear as children of the <definition> element. The content of each <target-class> element is an eDirectory class name specifying an object class, objects of which may be referred to by the eDirectory attribute specified by the attr-name attribute on the <definition> element. The absence of a <target-class> element indicates that any object is allowable.

Attributes

Attribute	Possible Values	Default Value

Example for Agent presentation for entry field with associated object selection button:

```
<definition
  type="dn-ref"
  name="pwd-policy"
  display-name="Password Policy object used for password generation"
  attr-name="nspmPasswordPolicyDN"
  aux-class-name="DirXML-PasswordGeneration">

  <target-class>nspmPasswordPolicy</target-class>
</definition>
```

Parent Elements

definition

See Table 8-1 for a list of method summary....

The following table shows the types used to represent the various GCV syntaxes.

GCV Methods

The following table contains a list of GCV methods:

Tables

Table 8-1 *GCV Methods*

Element	Description
clone ()	Creates and returns a copy of an object.
equals ()	Indicates whether some other object is "equal to" this one.
finalize ()	Called by the garbage collector on an object when garbage collection determines that there are no more references to the object.
getClass ()	Returns the runtime class of an object.
hashCode ()	Returns a hash code value for the object.
notify ()	Wakes up a single thread that is waiting on this object's monitor.
notifyAll ()	Wakes up all threads that are waiting on this object's monitor.
toString ()	Returns a string representation of the object.
wait ()	Causes current thread to wait until another thread invokes the notify() method or the notifyAll() method for this object.
wait (long timeout)	Causes current thread to wait until either another thread invokes the notify() method or the notifyAll() method for this object, or a specified amount of time has elapsed.
wait (long timeout, int nanos)	Causes current thread to wait until another thread invokes the notify() method or the notifyAll() method for this object, or some other thread interrupts the current thread, or a certain amount of real time has elapsed.

Table 8-2 *GCV Syntaxes*

Syntax	Type	Components (Notes)
VAL_STRING		
VAL_BOOLEAN		
VAL_INTEGER		
VAL_REAL		
VAL_DN		
VAL_ENUM		
VAL_PASSWORD_REF		
VAL_DN_REF		
VAL_LIST		
VAL_STRUCTURED		

Syntax	Type	Components (Notes)
VAL_LDAP		
VAL_SLASH		
VAL_QUALIFIED_SLASH		
VAL_DOT		
VAL_QUALIFIED_DOT		
VAL_CUSTOM		
VAL_DIRXML		
VAL_APPLICATION		
VAL_TRUE		
VAL_FALSE		
ATTR_NAME		
ATTR_DISPLAY_NAME		
ATTR_DISPLAY_NAME_REF		
ATTR_DESCRIPTION_REF		
ATTR_TYPE		
ATTR_DN_TYPE		
ATTR_DN_DELIMS		
ATTR_DN_SPACE		
ATTR_RANGE_LO		
ATTR_RANGE_HI		
ATTR_HIDE		
ATTR_ITEM_SEPARATOR		
ATTR_ATTR_NAME		
ATTR_AUX_CLASS_NAME		
ATTR_MULTILINE		
ATTR_TYPE_HINT		
ATTR_MANDATORY		
ATTR_DRIVER_PARAM_NAME		
ATTR_MIN_COUNT		
ATTR_MAX_COUNT		

Syntax	Type	Components (Notes)
ATTR_VALUE_SEPARATOR		
ATTR_INSTANCE_SEPARATOR		
TAG_CONFIGURATION_VALUES		
TAG_DEFINITIONS		
TAG_DEFINITION		
TAG_VALUE		
TAG_DESCRIPTION		
TAG_ENUM_CHOICE		
TAG_HEADER		
TAG_GROUP		
TAG_SUBORDINATES		
TAG_GCV_REF		
TAG_LIST		
TAG_ITEM		
TAG_TARGET_CLASS		
TAG_TEMPLATE		
TAG_INSTANCE		
ATTR_XML_SPACE		
VAL_PRESERVE		
EX_TAG_CONFIGURATION_VALUES		
EX_TAG_DEFINITION		
EX_TAG_VALUE		
EX_ATTR_NAME		
EX_ATTR_TYPE		
EX_ATTR_DISPLAY_NAME		

8.5 Configuration Value Type Usage

The following table shows which CV types are used by which CV implementations:

Value Types	GCVs	ECVs	Shim Parameters	Resource Parameters	Job Parameters
string	yes	yes	yes	yes	yes
boolean	yes	yes	yes	yes	yes
integer	yes	yes	yes	yes	yes
real	yes	yes	yes	yes	yes
dn	yes	yes	yes	yes	yes
enum	yes	yes	yes	yes	yes
list	yes	yes	yes	no	yes
structured	yes	no	yes	no	yes
password-ref	yes	no	yes	yes	yes

8.6 Type Usage

The following table shows which CV types are used by which CV implementations.

Value type	GCVs	ECVs	Shim Parameters	Resource Parameters	Job Parameters
string	yes	yes	yes	yes	yes
boolean	yes	yes	yes	yes	yes
integer	yes	yes	yes	yes	yes
real	yes	yes	yes	yes	yes
dn	yes	yes	yes	yes	yes
enum	yes	yes	yes	yes	yes
list	yes	yes	yes	no	yes
structured	yes	no	yes	no	yes
password-ref	no	no	yes	yes	yes
gcv-ref	no	no	yes	no	no
dn-ref	no	no	no	no	yes

8.7 Use of Global Configuration Values

GCVs are available to the driver policy and shim parameters. The actual values available are a combination of the following:

- GCV definitions on the DirXML-DriverSet object
- GCV definitions on the DirXML-Driver object
- Automatic GCVs

If there are any name collisions between the three sources the order of precedence is: driver object, driver set object, automatic. Automatic GCVs are values that are supplied by the Engine and provide information about the runtime environment. The automatic GCVs are:

Table 8-3 *Automatic GCVs*

Name	Type	Explanation
dirxml.auto.treename	string	The name of the eDirectory tree.
dirxml.auto.driverdn	string	The slash-form DN of the DirXML-Driver object.
dirxml.auto.driverguid	string	The value of the GUID attribute of the DirXML-Driver object in the following form: {xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx} where each 'x' is a hexadecimal digit.
dirxml.auto.localserverdn	string	The DN of the local server.

There are two basic ways to access a GCV from policy:

- Section 8.7.1, “Text Replacement,” on page 569
- Section 8.7.2, “DirXML-Script Access,” on page 570

8.7.1 Text Replacement

The following syntax is used to supply the value of a GCV via text replacement:

```
~gcv-name~
```

where "gcv-name" is the name of the desired GCV.

Text replacement works by replacing such references with the value of the GCV in policy source and in the following shim configuration fields:

Table 8-4 *Shim Configuration Fields with GCV Replacement*

Field	DirXML-Driver attribute	Explanation
shim auth id	DirXML-ShimAuthID	Typically presented by user agents as "Authentication ID", this value is usually used by a shim as the account name with which to authenticate to the target application.
shim auth server	DirXML-ShimAuthServer	Typically presented by user agents as "Authentication context", this value is usually used by a shim as connection information for the target application.

In policy source (both DirXML-Script and XSLT) the text replacement is straightforward and consists of simply replacing the reference with the string value of the GCV with one exception: List values will be output as either a string or as structured XML depending on the replacement context. In particular, if the list GCV reference appears in element content then the list will be output in a structured form. If the list GCV reference appears in an attribute value then the list will be output as a delimited string.

Several examples will help to illustrate text replacement (note the use of quotes around the GCV reference when the result needs to be a string):

Table 8-5 *GCV Text Replacement Example*

GCV Definition	`<definition`
	`name="my-gcv"`
	`type="string"`
	`display-name="Source subtree"`
	`>`
	`<value>\mytree\novell\</value>`
	`</definition>`

Reference	`<xsl:if test="starts-with(@src-dn,'~my-gcv~'">`
Result	`<xsl:if test="starts-with(@src-dn,'\mytree\novell\'">`

8.7.2 DirXML-Script Access

When the ~GCV~ notation is executed as a string replace at driver startup in XSLT or XPATH, the strings must be enclosed. However, in XPATH a single $GCV is sufficient, and need not be string enclosed.

In variable expansion supporting fields, $GCVName$ is sufficient and is not equal to ~GCVName~ as the $GCVName$ expansion is executed at run time as it is encountered, but ~GCVName~ is evaluated at driver start time.

See the DirXML Script documentation (http://developer.novell.com/documentation/dirxml/dirxmlbk/ref/dirxmlscript/index.html) for the full description. GCVs can be accessed in DirXML Script policies in several ways:

- Text Replacement
- XPath Variables: XPath expressions in DirXML-Script can reference GCVs through the standard XPath "$name" syntax for variables. For more information, see XPath Variables documentation (http://www.w3.org/MarkUp/Forms/wiki/Variables_in_XPath).
- Conditions and Tokens: The `<if-global-variable>` condition and the `<token-global-variable>` token. For more information, see DirXMLScript DTD (http://developer.novell.com/documentation/dirxml/dirxmlbk/ref/dirxmlscript/index.html).
- Variable Expansion: Many conditions and tokens support the DirXML Script variable expansion where the GCV is referenced as $gcv-name$. For more information, see DirXMLScript DTD (http://developer.novell.com/documentation/dirxml/dirxmlbk/ref/dirxmlscript/index.html).

9 DS-Object DTD

The ds-object DTD is used to create an object of the specified class and name in the container object. You can specify the hierarchy of objects to be created as part of driver import in a driver configuration file. The specification of these objects takes the form of ds-object elements as first level children of the driver-configuration element. A ds-object construct is also used for importing provisioning objects within a provisioning element, Role-based Entitlement objects within an rbe-policies element, and Identity Manager Job objects within a Jobs element.

The driver configuration level ds-object elements are processed after all variable analysis and patching is completed so that the contents of the ds-object elements can be controlled by user prompts in the driver configuration file. The top level ds-object element specifies an object to be created at the root of the tree being imported into. To create object's within a nested container, you must specify the hierarchy of the objects using the nested ds-object elements that specify the appropriate class for each object in the hierarchy. If your driver configuration file creates the objects in the hierarchy, you need to include appropriate ds-attribute elements to specify the attributes of the these objects.

The ds-object elements are used for object creation only. If an object by the same name and class name, such as ads-object, already exists in the Identity Vault, the object is not changed (that is, the ds-attribute values from the driver configuration file are not applied). If an object with the same name from a different class exists, an error is generated. The ds-object child elements of ds-object elements are always processed using the same rules. The following is an example of ds-object XML definition:

```
<add class-name="User" src-dn="\Sam">
 <association>1012</association>
 <add-attr attr-name="cn">
  <value>Sam</value>
 </add-attr>
 <add-attr attr-name="Surname">
  <value>Jones</value>
 </add-attr>
 <add-attr attr-name="Given Name">
  <value>Sam</value>
 </add-attr>
 <add-attr attr-name="Telephone Number">
  <value>555-1212</value>
 </add-attr>
</add>
```

9.1 DS-Object DTD Elements

Element	Description
ds-object	Creates an object of the specified class and name in the container object.
ds-attributes	Adds attributes on the containing ds-object element.

Element	Description
ds-attributes (job)	Adds job specific values and query information defining a job on the containing ds-object element.
ds-aux-class-attributes	Adds attributes on an auxiliary class on the containing ds-object element.
ds-rights-other-objects	Adds information about an Role-based Entitlements policy's rights to other objects.
ds-rights-object	Adds object information about an Role-based Entitlements policy's rights.
ds-rights-attribute	Adds attribute information about an Role-based Entitlements policy's rights.
ds-attribute	Adds an attribute and value(s) to be stored on the containing ds-object.
ds-member-query-url-info	Adds an LDAP memberQueryURL attribute value.
ds-value	Adds individual attribute values for an attribute.

ds-object

Creates an object of class specified in the ds-object-class attribute in the container object. The ds-object element describes an object hierarchy that is to be created in the Identity Vault as part of importing the driver configuration file. When contained in a provisioning element, the object is created in the DirXML-Driver object. When nested in another ds-object element, the object is created under that object. The containing ds-object element must specify an Identity Vault container class that can contain an object of the specified class. When contained in a driver-configuration element, the object is created at the root of the eDirectory tree. If the object already exists, the attributes are ignored but any nested ds-object's are processed.

Attributes

Attribute	Possible Values	Default Value
ds-object-class	CDATA The class name of the object to create in the Identity Vault.	#REQUIRED
ds-object-name	CDATA The name of the class under which the object is to be to created in the Identity Vault.	#REQUIRED
base-dn	CDATA The base dn from the root where the object is to created in the Identity Vault.	#REQUIRED
on-update	CDATA The possible value is “overwrite” and the default value is “ignore”. overwrite can be used to indicate overwriting existing values on an eDir object. By default, it is set to “ignore.” If an object mentioned in the ds-object is already present in Identity Vault, ds-object creation does not perform anything. This option can only be used if you are deploying ds-objects through Designer.	None
Attribute	Possible Values	Default Value
ds-object-class	CDATA The class name of the object to create in the Identity Vault.	#REQUIRED

Allowed Content

Attribute	Possible Values
ds-object	Adds object within this object.
ds-attributes	Adds attributes to the containing ds-object.
ds-attributes (job)	Adds the job specific attributes and queries information on the containing ds-object defining a job.
ds-aux-class-attributes	Adds attributes of the auxiliary class on the containing ds-object.
ds-rights-other-objects	Adds ds-object defines an Role-based Entitlement policy.

Content Rule

(ds-object, ds-attributes, ds-aux-class-attributes, ds-rights-other-objects)

Parent Elements

Element	Description
ds-object	Creates an object of the specified class and name in the container object.

ds-attributes

Contains attributes to be stored on an auxclass of the containing ds-object.

Allowed Content

Element	Description
ds-attribute	Contains an attribute and value(s) to be stored on the containing ds-object.

Attributes

None

Content Rule

(ds-attribute*)

Parent Elements

Element	Description
ds-object	Create an object of specified class and name in the container object.

ds-attributes (job)

Used within a ds-object defining a job, contains attributes to be stored on the job as well as job specific values and queries.

Allowed Content

Element	Description
job-email-server-query	Contains the the email server to be associated with the containing DirXML-Job.
job-scope-query	Contains the scope information for the job.
job-servers-query	Contains the servers to be associated with the job.
job-reference-dnquery	Contains a job reference DN for the job. The number and content of these is based on the content of the exported job's XmlData.

Attributes

Attribute	Possible Values	Default Value
jjob-name	CDATA	#REQUIRED
job-display-name	CDATA	#IMPLIED
attr-name	CDATA	#REQUIRED
aux-class-name	CDATA	#IMPLIED
reference-name	CDATA	#REQUIRED
reference-display-name	CDATA	#IMPLIED
target-class-filter	CDATA	#IMPLIED

Content Rule

(job-email-server-query, job-scope-query, job-servers-query, (job-reference-dn-query)

Parent Elements

Element	Description
ds-object	Create an object of specified class and name in the container object.

ds-aux-class-attributes

Contains attributes to be stored on an auxclass on the containing ds-object.

Allowed Content

Element	Description
ds-attribute	Contains an attribute and value(s) to be stored on the containing ds-object.

Attributes

Attribute	Possible Values	Default Value
aux-class-name	CDATA Name of the auxclass the contained ds-attribute.	#REQUIRED

Content Rule

(ds-attribute*)

Parent Elements

Element	Description
ds-object	Creates an object of specified class and name in the container object.

ds-rights-other-objects

Contains information about an RBE policy's rights to other objects.

Allowed Content

Element	Description
ds-rights-object	Adds object information about an Role-based Entitlements policy's rights.

Attributes

None

Content Rule

(ds-rights-object*)

Parent Elements

Element	Description
ds-object	Creates an object of specified class and name in the container object.

ds-rights-object

Adds object information about an RBE policy's rights.

Allowed Content

Element	Description
ds-rights-attribute	Adds attribute information about an RBE policy's rights.

Attributes

Attribute	Possible Values	Default Value
dn		#REQUIRED

Content Rule

(ds-rights-attribute*)

Parent Elements

Element	Description
ds-rights-other-objects	Contains information about an RBE policy's rights to other objects.

ds-rights-attribute

Adds attribute information about an RBE policy's rights.

Allowed Content

None

Attributes

Attribute	Possible Values	Default Value
ds-attr-name	CDATA	#REQUIRED
ds-rights	Attribute name for these rights	#REQUIRED
	CDATA	
	The rights for this attribute	

Content Rule

None

Parent Elements

Element	Description
ds-rights-objects	Adds object information about an RBE policy's rights.

ds-attribute

Contains an attribute and value(s) to be stored on the containing ds-object. If there are multiple ds-values, the attribute named by ds-attr-name must be defined in eDirectory to contain multiple values.

Allowed Content

Element	Description
ds-value	Add attribute value
ds-member-query-url-info	used with the DirXML-SPFilterXML of a DirXML-SharedProfile

Attributes

Attribute	Possible Values	Default Value
ds-attr-name	CDATA	#REQUIRED

Content Rule

(ds-value*, ds-member-query-url-info?)

Parent Elements

Element	Description
ds-attributes	Adds attributes on the containing ds-object element.
ds-aux-class-attributes	Adds attributes on an auxiliary class on the containing ds-object element.

ds-member-query-url-info

Contains information used to construct an LDAP memberQueryURL attribute value from a DirXML-SPFilterXML value. Used only with the DirXML-SPFilterXML attribute of a DirXML-SharedProfile --object when exporting an RBE policy.

Allowed Content

None

Attributes

Attribute	Possible Values	Default value
base-dn	CDATA The base DN of the query in Fully Qualified Distinguished Name format.	#REQUIRED
scope	CDATA The scope of the query, such as base, one, sub, and so on. Default value is “base”.	#IMPLIED
x-chain	CDATA The x-chain of the query. Default value is “”.	#IMPLIED

Content Rules

None

Parent Elements

Element	Description
ds-attribute	Contains an attribute and value(s) to be stored on the containing ds-object.

ds-value

Contains individual attribute values for an attribute. The contents are based on the syntax of the attribute. The stream attributes are stored as base64 encoded strings.

Allowed Content

None

Attributes

Attribute	Possible Values	Default Value
base64-encoded	"true" or “false”	false
contains	"text", "base64encoded", or "xml"	None

Content Rule

None

Parent Elements

Element	Description
ds-attribute	Contains an attribute and value(s) to be stored on the containing ds-object.

10 EntitlementConfiguration DTD

Entitlements are a way for you to provide users with access to resources in the connected systems. Entitlements allow you to store parametrized flags on objects in the Identity Vault. The Identity Manager drivers implement entitlements and based on the entitlement flags, add or remove users from a role or a group.

An EntitlementConfiguration object contains meta-data about the entitlements defined for any Identity Manager driver. The entitlementconfiguration object has been standardized/extended in Identity Manager 4.0. It introduces a common format and provides additional extensions that can be used by the Identity Reporting module for the data collection service.

The <entitlement-configuration> node contains the metadata about the various entitlements for a driver.

```
<entitlement-configuration modified="20121004122936">
<entitlements>
    <entitlement resource-mapping-state="add" parameter-format="legacy"
dn="CN=ExchangeMailbox,CN=Active
Directory,CN=driverset1,dc=idm,dc=services,dc=system" resource-mapping="true"
role-mapping="true">
        <type id="mailbox" name="mailbox" category="other account">
            <display-name>
                <value langCode="EN">Mailbox</value>
            </display-name>
        </type>
    </entitlement>
    <entitlement resource-mapping-state="pending" parameter-format="idm4"
dn="CN=Group,CN=Active Directory,CN=driverset1,dc=idm,dc=services,dc=system"
resource-mapping="true" role-mapping="true">
        <type id="group" name="group" category="security grouping">
            <display-name>
                <value langCode="EN">Group</value>
            </display-name>
        </type>
        <parameters>
            <parameter mandatory="true" name="ID" source="read-attr" source-
name="ID"/>
            <parameter mandatory="true" name="ID2" source="src-dn"/>
        </parameters>
        <member-assignment-extensions>
            <query-attr name="query-type">entitlement-assignment</query-attr>
            <query-xml>
                <read-attr attr-name="member"/>
            </query-xml>
        </member-assignment-extensions>
        <query-extensions>
            <query-attr name="extension-type">data</query-attr>
            <query-xml>
                <read-attr attr-name="owner"/>
                <read-attr attr-name="sAMAccountName"/>
            </query-xml>
        </query-extensions>
    </entitlement>
    <entitlement dn="CN=UserAccount,CN=Active
Directory,CN=driverset1,dc=idm,dc=services,dc=system" resource-mapping="true"
role-mapping="true">
```

```
        <type id="user" name="account" category="security account">
            <display-name>
                <value langCode="EN">User</value>
            </display-name>
        </type>
        <member-assignment-query>
            <query-attr name="query-type">entitlement-assignment</query-attr>
            <query-xml>
                <nds dtdversion="2.0">
                    <input>
                        <query class-name="User" scope="subtree">
                            <search-class class-name="User"/>
                            <read-attr/>
                        </query>
                    </input>
                </nds>
            </query-xml>
        </member-assignment-query>
        <query-extensions>
            <query-attr name="extension-type">accounts</query-attr>
            <query-xml>
                <read-attr attr-name="dirxml-uACAccountDisable"/>
                <read-attr attr-name="userPrincipalName"/>
                <read-attr attr-name="sAMAccountName"/>
            </query-xml>
        </query-extensions>
        <account>
            <account-id source="read-attr" source-name="sAMAccountName"/>
            <account-id source="read-attr" source-name="userPrincipalName"/>
            <account-id source="src-dn"/>
            <account-id source="association"/>
            <account-status source="read-attr" source-name="userAccountControl"
active="false" inactive="true"/>
        </account>
    </entitlement>
</entitlements>
</entitlement-configuration>
```

10.1 DirXML EntitlementConfiguration DTD Elements

Element	Description
account	Gets account information.
account-id	Account ID in the connected system.
account-status	Account status (active/inactive) in connected system.
connection	Logical connection.
connections	Logical connections for fan-out configuration.
display-name	Display names for different languages.
entitlement	Entitlement definition.
entitlement-configuration	Entitlement configuration.
entitlements	Entitlement list.
filter	Filter for query results.
filters	Filters to include or exclude query results.
member-assignment-extensions	Member assignment information for groups.

Element	Description
member-assignment-query	Member assignment information for accounts.
parameter	Entitlement parameter definition.
parameters	Entitlement parameter list.
query-attr	Query attributes to add to the base query.
query-extensions	Optional query extensions.
query-instance	Entitlement query.
query-xml	Modifications for queries.
sub-type	Entitlement sub-type.
type	Entitlement type.
value	Holds values.

account

It is applicable only for account entitlement. The purpose of this extension is to instruct the querying agent how to interpret the results of a <member-assignment-query> of an <entitlement> with type name <account>. The extensions are stored in a new <account> child node of the <entitlement> node in the entitlements section. The <account> node groups the <account> extension instructions.

Example

```
<account>
  <account-id source="read-attr" source-name="sAMAccountName"/>
  <account-id source="read-attr" source-name="userPrincipalName"/>
  <account-id source="src-dn"/>
  <account-id source="association"/>
  <account-status source="read-attr" source-name="dirxml-uACAccountDisable"
active="false" inactive="true"/>
</account>
```

Allowed Content

Element	Description
account-id	The account ID in the connected system.
account-status	The active/inactive status of the account in the connected system.

Attributes

None

Content Rule

(account-id + , account-status ?)

Parent Elements

Element	Description
entitlement	Entitlement definition.

account-id

Informs the querying agent where it can obtain an ID of the <account> in the managed (connected) system. An <account> node can have multiple <account-id> nodes.

Example

See <account>.

Allowed Content

Any

Attributes

Attribute	Value(s)	Default Value
source	read-attr | search-attr | src-dn | association | external Indicates the source from where the data must be taken from.	#REQUIRED
source-name	CDATA Name for the source type.	#IMPLIED

Content Rule

Any

Parent Elements

Element	Description
account	Get account information.

account-status

Instructs the querying agent which attribute indicates the <account> status (enabled/disabled) and what the syntax is. The attributes on the <account-id> and <account-status> nodes work similar to the attributes on the <parameter> node.

Example

See <account>.

Allowed Content

Any

Attributes

Attribute	Value(s)	Default Value
active	CDATA	#REQUIRED
	The value indicating the account is active.	
inactive	CDATA	#REQUIRED
	The value indicating the account is inactive.	
source	read-attr | search-attr | src-dn | association | external	#REQUIRED
	Indicates the source from where the data must be taken from.	
source-name	CDATA	#REQUIRED
	The attribute name for the source type.	

Content Rule

Any

Parent Elements

Element	Description
account	Get account information.

connection

Specifies a <connection> (logical system). This is only applicable for the fan-out drivers.

Example

See <connections>.

Allowed Content

Element	Description
query-attr	Query attributes to add to the base query.
query-xml	Modifications for queries.
query-instance	Entitlement query.

Attributes

Attributes	Value(s)	Default Value
name	CDATA	#REQUIRED
	Connection identifier.	

Content Rule

(query-attr * , query-xml * , query-instance *)

Parent Elements

Element	Description
connections	Logical connections for the fan-out configuration.

connections

List of <connections> (logical systems for the fan-out configuration used in the driver configuration, such as SAP driver).

Example

```
<connections>
  <connection name="ADMCLNT811">
    <query-attr name="dest-dn">\ADMCLNT811</query-attr>
    <query-instance langCode="en">
        <query-xml>
            <search-attr attr-name="LANGCODE">
                <value>en</value>
            </search-attr>
        </query-xml>
    </query-instance>
</connection>
</connections>
```

Allowed Content

Element	Description
connection	Logical connection.

Attributes

None

Content Rule

(connection)

Parent Elements

Element	Description
entitlement-configuration	Entitlement configuration.

display-name

Specifies the <display-name> for various languages.

Example

```
<display-name>
  <value langCode="EN">Mailbox</value>
</display-name>
```

Allowed Content

Element	Description
value	For holding values.

Attributes

None

Content Rule

(value*)

Parent Elements

Element	Description
sub-type	Entitlement sub-type.
type	Entitlement type.

entitlement

Defines an <entitlement>. It contains the dn of the actual <entitlement> definition that is stored in the XMLData attribute in the Identity Vault. This element contains additional information about the <entitlement> type, filters, and additional extensions that can be used by clients for data collection.

Example

See <entitlement-configuration>.

Allowed Content

Element	Description
type	Entitlement type.
parameters	Parameter list.
filters	Filters to include or exclude query results.
query-xml	Modifications for queries.
member-assignment-query	Member assignment information for accounts
member-assignment-extensions	Member assignment information for groups.
query-extensions	Optional query extensions.
account	Get account information.

Attributes

Attribute	Value(s)	Default Value
dn	CDATA	#REQUIRED
	The LDAP DN of the entitlement object in the Identity Vault.	
name	CDATA	#IMPLIED
	The Identity Vault name of the entitlement.	
parameter-format	legacy | idm4	idm4
	Indicates the format for the parameters. The parameter formats before Identity Manager 4.0 are referred as legacy formats.	
resource-mapping	true | false	true
	Indicates whether this entitlement should be included in the resource mapping.	
resource-mapping-state	add | pending	add
	Whether resource associations should be added with immediate effect or wait for approval.	

Attribute	Value(s)	Default Value
role-mapping	true | false Indicates whether this entitlement should be included in the role mapping.	true

Content Rule

(account, type, parameters, filters, query-extensions, query-xml, member-assignment-extensions, member-assignment-query)

Parent Elements

Element	Description
entitlement	Interpretation of the query results.

entitlement-configuration

Contains the metadata of various entitlements for a driver.

Example

```
<entitlement-configuration modified="20121004122936">
<entitlements>
    <entitlement resource-mapping-state="add" parameter-format="legacy"
dn="CN=ExchangeMailbox,CN=Active
Directory,CN=driverset1,dc=idm,dc=services,dc=system" resource-mapping="true"
role-mapping="true">
        <type id="mailbox" name="mailbox" category="other account">
            <display-name>
                <value langCode="EN">Mailbox</value>
            </display-name>
        </type>
    </entitlement>
    <entitlement resource-mapping-state="pending" parameter-format="idm4"
dn="CN=Group,CN=Active Directory,CN=driverset1,dc=idm,dc=services,dc=system"
resource-mapping="true" role-mapping="true">
        <type id="group" name="group" category="security grouping">
            <display-name>
                <value langCode="EN">Group</value>
            </display-name>
        </type>
        <parameters>
            <parameter mandatory="true" name="ID" source="read-attr" source-
name="ID"/>
            <parameter mandatory="true" name="ID2" source="src-dn"/>
        </parameters>
        <member-assignment-extensions>
            <query-attr name="query-type">entitlement-assignment</query-attr>
            <query-xml>
                <read-attr attr-name="member"/>
            </query-xml>
        </member-assignment-extensions>
        <query-extensions>
            <query-attr name="extension-type">data</query-attr>
            <query-xml>
                <read-attr attr-name="owner"/>
                <read-attr attr-name="sAMAccountName"/>
            </query-xml>
        </query-extensions>
    </entitlement>
    <entitlement dn="CN=UserAccount,CN=Active
Directory,CN=driverset1,dc=idm,dc=services,dc=system" resource-mapping="true"
role-mapping="true">
        <type id="user" name="account" category="security account">
            <display-name>
                <value langCode="EN">User</value>
            </display-name>
        </type>
        <member-assignment-query>
            <query-attr name="query-type">entitlement-assignment</query-attr>
            <query-xml>
                <nds dtdversion="2.0">
                    <input>
                        <query class-name="User" scope="subtree">
                            <search-class class-name="User"/>
                            <read-attr/>
                        </query>
                    </input>
                </nds>
            </query-xml>
        </member-assignment-query>
        <query-extensions>
```

```
            <query-attr name="extension-type">accounts</query-attr>
            <query-xml>
                <read-attr attr-name="dirxml-uACAccountDisable"/>
                <read-attr attr-name="userPrincipalName"/>
                <read-attr attr-name="sAMAccountName"/>
            </query-xml>
        </query-extensions>
        <account>
            <account-id source="read-attr" source-name="sAMAccountName"/>
            <account-id source="read-attr" source-name="userPrincipalName"/>
            <account-id source="src-dn"/>
            <account-id source="association"/>
            <account-status source="read-attr" source-name="userAccountControl"
active="false" inactive="true"/>
        </account>
    </entitlement>
</entitlements>
</entitlement-configuration>
```

Allowed Content

Element	Description
entitlements	Entitlement list.
connections	Logical connections for fan-out configuration.

Attributes

Attribute	Value(s)	Default Value
driver-type	CDATA The driver type.	#IMPLIED
min-driver-version	CDATA Indicates the minimum version of the driver shim. This should only be used in conjunction with driver-type to indicate to the UI agent which driver and version are expected for the configuration to work.	#IMPLIED
modified	CDATA The date or time when the configuration file was modified.	#REQUIRED
rbpm-refresh-rate	CDATA The RBPM configuration parameter.	#IMPLIED
rbpm-time-out	CDATA The RBPM configuration parameter.	#IMPLIED

Content Rule

(entitlements ?, connections ?)

Parent Elements

None

entitlements

Lists <entitlements> used in the driver configuration.

Example

See <entitlement-configuration>.

Allowed Content

Element	Description
entitlement	Entitlement definition.
token-src-dn	Uses association value from the query result.
token-attr	Uses attribute value from the query result.

Content Rule

(token-association | token-src-dn | token-attr)

Parent Elements

Element	Description
result-set	Interpretation of the query results.

filter

Defines a filter to apply to the query results.

Example

See <entitlement> filters.

Allowed Content

#PCDATA

Content Rule

#PCDATA

Attributes

Attribute	Value(s)	Default Value
regex	CDATA	#REQUIRED
	The regular expression to apply to the query result.	
source	read-attr | src-dn | association	#REQUIRED
	The source of the value to which the regex should be applied.	
source-name	CDATA	#REQUIRED
	The attribute name for the READ_ATTR source type.	

Parent Elements

Element	Description
filters	Filters to include or exclude the query results.

filters

Defines the list of filters to include or exclude the query results.

Example

```
<filters type="exclude">
  <filter regex="^&.+" source="read-attr" source-name="BAPIPROF"/>
</filters>
```

Allowed Content

Element	Description
token-association	Filter for query results.
token-src-dn	Use association value from the query result.
token-attr	Use attribute value from the query result.

Attributes

Attribute	Value(s)	Default Value
type	include | exclude	include
	The type of the filters, whether to include or exclude from the query result.	

Content Rule

(filter)

Parent Elements

Element	Description
entitlement	Entitlement definition.

member-assignment-extensions

It is applicable to the <entitlement> type group only. The Identity Manager Data Collection Service (DCS) should be able to obtain the member assignment information from the managed and unmanaged accounts. Because member assignment information can get bulky when the groups and roles have many assigned members, the query extensions to retrieve this information are optional. Appropriate query extensions should be added to the querying agent. The member assignment information is available on both account and group entitlements but it must be retrieved slightly differently. For entitlement with the `group` type name, the content of the query-xml child node of the member-assignment-query node is expected to be an extension to the existing query defined in the entitlement XML. For example, search-attr or read-attr nodes.

Example

```
<member-assignment-extensions>
  <query-attr name="query-type">entitlement-assignment</query-attr>
  <query-xml>
    <read-attr attr-name="member"/>
  </query-xml>
</member-assignment-extensions>
```

Allowed Content

Element	Description
query-app	Query attributes to add to the base query.
query-xml	Modifications for queries.

Attributes

None

Content Rule

(query-app *, query-xml *)

Parent Elements

Element	Description
entitlement	Entitlement definition.

member-assignment-query

It is applicable for <entitlement> type account only. The Identity Manager Data Collection Service (DCS) should be able to obtain the member assignment information from the managed and unmanaged accounts. Because member assignment information can get bulky when the groups and roles have many assigned members, the query extensions to retrieve this information are optional. Appropriate query extensions should be added to the querying agent. Member assignment information is available on both account and group entitlements but it must be retrieved slightly differently. For entitlement with account `type name`, the content of the query-xml child node of the member-assignment-query node is expected to be a full query definition that replaces the potentially existing query defined in the entitlement XML.

Example

```
<member-assignment-query>
  <query-attr name="query-type">entitlement-assignment</query-attr>
  <query-xml>
    <nds dtdversion="2.0">
      <input>
        <query class-name="User" scope="subtree">
          <search-class class-name="User"/>
          <read-attr/>
        </query>
      </input>
    </nds>
  </query-xml>
</member-assignment-query>
```

Allowed Content

Element	Description
query-attr	Query attributes to add to the base query.
query-xml	Modifications for queries.

Attributes

None

Content Rule

(query-attr *, query-attr *)

Parent Elements

Element	Description
entitlement	Entitlement definition.

parameter

Declares a component of the parameter value.

Example

See <parameters>.

Allowed Content

#PCDATA

Attributes

Attribute	Value(s)	Default Value
mandatory	true | false	true
	Indicates whether the parameter is mandatory.	
name	CDATA	#REQUIRED
	The name component of the parameter name/value pair.	
source	read-attr | search-attr | src-dn | association | external | connection	#REQUIRED
	Indicates where the data for the parameter value must be taken from.	
source-name	CDATA	#IMPLIED
	Attribute name for source type.	

Content Rule

(#PCDATA)

Parent Elements

Element	Description
parameters	Parameters list.

parameters

Declares a component of the parameter value.

Example

```
<parameters>
  <parameter mandatory="true" name="ID" source="read-attr" source-name="ID"/>
  <parameter mandatory="true" name="ID2" source="src-dn"/>
</parameters>
```

Allowed Content

Element	Description
parameter	Parameter definition.

Attributes

None

Content Rule

parameter

Parent Elements

Element	Description
entitlement	Entitlement definition.

query-attr

Contains the XML attribute/value to add to the base entitlement query element.

Example

```
<query-attr name="query-type">entitlement-assignment</query-attr>
```

Allowed Content

#PCDATA

Attributes

Attribute	Value(s)	Default Value
name	CDATA	#REQUIRED
	The name of the attribute to add to the entitlement query.	

Content Rule

(#PCDATA)

Parent Elements

Element	Description
connection	A logical connection.
member-assignment-extensions	Member assignment information for groups.
member-assignment-query	Member assignment information for accounts.
query-extensions	Optional query extensions.
query-instance	Entitlement query.

query-extensions

The Identity Manager Data Collection Service should be able to obtain additional information from the managed systems than what is defined on the base query and the extensions for role and resource mapping. Therefore, the query extensions to retrieve this information are optional and have to be added by the querying agent on an as-needed basis. The query-extensions are supported on all entitlements.

Example

```
<query-extensions>
  <query-attr name="extension-type">data</query-attr>
  <query-xml>
    <read-attr attr-name="owner"/>
    <read-attr attr-name="sAMAccountName"/>
  </query-xml>
</query-extensions>
```

Allowed Content

Element	Description
query-attr	Query attributes to add to the base query.
query-xml	Modifications for queries.

Attributes

None

Content Rule

(query-attr*, query-xml*)

Parent Elements

Element	Description
entitlement	Entitlement definition.

query-instance

Contains the XML element indicating a separate entitlement query.

Example

```
<query-instance langCode="en">
  <query-xml>
    <search-attr attr-name="LANGCODE">
      <value>en</value>
    </search-attr>
  </query-xml>
</query-instance>
```

Allowed Content

Element	Description
query-xml	Query attributes to add to the base query.
query-attr	Modifications for queries.

Attributes

Attribute	Value(s)	Default Value
langcode	CDATA	#REQUIRED
	The two character language code.	

Content Rule

(query-attr*, query-xml *)

Parent Elements

Element	Description
connection	A logical connection.

query-xml

Contains XML fragment to be added to the base query element.

Example

See <query-instance>.

Allowed Content

Any

Attributes

None

Content Rule

(Any)

Parent Elements

Element	Description
connection	A logical connection.
entitlement	Entitlement definition.
member-assignment-extensions	Member assignment information for groups.
member-assignment-query	Member assignment information for accounts.
query-extensions	Optional query extensions.
query-instance	Entitlement query.

sub-type

Contains a sub-type for the given entitlement. For example, a SAP role may be a 'composite' or a 'single' role.

Example

```
<sub-type source="read-attr" source-name="AGR_TYPE">
  <display-name source-value="C">
    <value langCode="en">Composit Role</value>
  </display-name>
  <display-name source-value="S">
    <value langCode="en">Single Role</value>
  </display-name>
</sub-type>
```

Allowed Content

Element	Description
display-name	Display names for different languages.

Attributes

Attribute	Value(s)	Default Value
source	read-attr | search-attr | src-dn | association | external The source of the sub-type value.	#REQUIRED
source-name	CDATA Attribute name for sub-type source attribute. It is used only when source is a read or search attribute.	#REQUIRED

Content Rule

(display-name)

Parent Elements

Element	Description
type	Entitlement type.

type

Defines an entitlement type.

Example

```
<type id="user" name="account" category="security account">
  <display-name>
    <value langCode="EN">User</value>
  </display-name>
</type>
```

Allowed Content

Element	Description
display-name	Display names for different languages.
sub-type	Entitlement sub-type.

Attributes

Attribute	Value(s)	Default Value
category	CDATA The category helps grouping different entitlements with the same type name into categories. For example, Active Directory supports a variety of different types of groupings: local groups, security groups, and distributions lists. All are of `group` type name, but from different categories, such as security grouping and other grouping.	#REQUIRED
id	CDATA The ID (id) identifies a type uniquely per managed system. There cannot be two entitlements with the same type ID defined for the same managed system (driver).	#REQUIRED
name	account | group | other The name provides a non-localized identifier for the type. Typically, the name should resemble or be equal to the English display name of type. Type names don't have to be unique for each managed system. There can be two or more entitlements with the same type name defined for the same managed system (driver).	#REQUIRED

Content Rule

(display-name ? , sub-type ?)

Parent Elements

Element	Description
entitlement	Entitlement definition.

value

Defines a value element. It can hold any value depending on the context.

Example

```
<value langCode="EN">User</value>
```

Allowed Content

#PCDATA

Attributes

Attribute	Value(s)	Default Value
langcode	CDATA	#REQUIRED
	The two ISO language code (ie "en", "de").	

Content Rule

(#PCDATA)

Parent Elements

Element	Description
display-name	The display names for different languages.